

BIBLIOTECA PC

3

FUNCȚII SISTEM



*micro*Informatica

Editat sub egida Comisiei Naţionale de Informatică

Sistemul de operare DOS*
Funcţii sistem

Ediţia III completată

Cluj-Napoca
1991

* MS-DOS este marcă înregistrată a firmei Microsoft
* PC-DOS este marcă înregistrată a firmei IBM

Autor

Vlad Căprariu

Tehnoredactor

Smaranda Derveşteanu

Coperta colecţiei

designer Liviu Derveşteanu

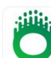
Editor

prof. univ. dr. Emil Muntean

Tehnoredactare computerizată şi pregătire pentru offset

MicroInformatica SRL

Cuprins



1. Apeluri de funcţii sistem

Rutinele utilizate de DOS pentru tratarea operaţiilor şi resurselor sistemului pot fi apelate de orice program de aplicaţii, uşurînd munca de programare şi mărind posibilitatea compatibilităţii cu viitoare versiuni DOS.

Rutinele DOS sînt apelate prin întreruperi software. Întreruperea 21h (INT 21h) este serviciul pentru apelul funcţiilor. Ea permite accesul la o mare varietate de resurse DOS.

Pentru un apel al unei funcţii sistem DOS, se procedează astfel:

- se încarcă numărul funcţiei în registrul AH;
- se încarcă (dacă este necesar) numărul subfuncţiei în registrul AL;
- se încarcă (dacă este necesar) celelalte date în regiştrii specificaţi;
- se generează INT 21h.

Exemplu:

```
mov ah, 02h
mov dl, 41h
int 21h
```

Unele funcţii necesită ca parametri şiruri de caractere ASCII (de exemplu, nume de fişiere, căi, etc). În acest caz, şirul trebuie să aibă contorul binar 0 la sfîrşit (terminator este caracterul nul), iar ca parametru se transmite adresa şirului.

Numărul funcţiilor a crescut o dată cu versiunile sistemului de operare, existînd în prezent şi funcţii cu acţiuni similare, dar cu

numere diferite. În aceste cazuri, se recomandă utilizarea celei cu număr mai mare.

Pentru înțelegerea acestor, funcții care în parte se acoperă, reamintim pe scurt istoricul sistemului de operare DOS.

Dezvoltarea sistemului s-a făcut în trei pași mari:

1. Primele versiuni (1.x) se apropiau foarte mult de sistemul de operare CP/M, din motive de compatibilitate necesară în acel moment.

2. Versiunile următoare (2.x) au adus structura de fișiere asemănătoare sistemului UNIX, și o tratare consecventă a erorilor, relativ la care se folosește următoarea convenție:

   - dacă operația cerută s-a executat corect, indicatorul CARRY (CF) va fi nul;

   - în caz de eroare, indicatorul CARRY este poziționat și se returnează un cod de eroare în registrul AX.

3. Începînd cu versiunea a treia, s-au adăugat funcții pentru lucrul cu rețele, cu fișiere partajate, funcții noi pentru gestionarea proceselor, precum și o funcție importantă pentru tratarea erorilor (59h), care oferă informații exacte despre originea unei (eventuale) erori la ultima funcție apelată, precum și sugestii (codificate) referitor la posibilitățile de tratare a erorii.

Cea mai importantă diferență între funcțiile vechi și cele noi constă în modul de tratare al fișierelor.

În cazul funcțiilor vechi, programatorul trebuie să pregătească un Bloc de Control al Fișierului (FCB - File Descriptor Block). Funcțiile noi necesită pentru tratarea fișierelor doar adresa numelui de fișier, returnînd o valoare numită identificator logic de fișier (handle), prin care acesta va fi identificat în operațiile de intrare/ieșire.

Utilizarea funcţiilor vechi prezintă avantaje şi dezavantaje:

- Pot fi citite direct din FCB anumite informaţii, care în cazul funcţiilor noi pot fi preluate doar prin apeluri de funcţii (nu se cunoaşte FCB-ul, ci numai un număr logic al fişierului). Acesta este motivul pentru care se mai utilizează această metodă, dar firmele Microsoft şi IBM nu încurajează acest mod de lucru.

- Utilizatorul trebuie să rezerve spaţiu de memorie pentru FCB şi să completeze cîmpurile lui. Această muncă nu mai este necesară la funcţiile noi.

Cel mai mare avantaj al noilor funcţii îl constituie posibilitatea redirectării fişierelor standard:

- intrare standard (identificator logic 0);
- ieşire standard (identificator logic 1);
- eroare standard (identificator logic 2).

Funcţiile sistem pot fi apelate şi dintr-un limbaj de nivel înalt, ale cărui module pot fi legate cu module scrise în limbaj de asamblare. În plus, majoritatea limbajelor de nivel înalt au prevăzute rutine de bibliotecă pentru execuţia directă a acestor funcţii.

Observaţii:

1. Cînd DOS preia controlul după un apel de funcţie, se face un salt la o stivă internă, salvîndu-se toţi regiştrii neutilizaţi pentru returnare de informaţii (cu excepţia registrului AX). Ca urmare, stiva programului apelant trebuie să fie suficient de mare pentru a se adapta sistemului de întreruperi (cel puţin 128 de octeţi în plus, faţă de alte cerinţe).

2. În capitolul 2 sînt tratate funcţiile sistem ale versiunii DOS 3.20, iar în anexe sînt prezentate completările aduse de versiunile DOS 3.30 şi 4.00, precum şi unele detalii despre funcţiile nedocumentate.

1.1. Apelurile de funcţii DOS, în ordine numerică

Funcţia 00h	-	Terminarea programului
Funcţia 01h	-	Citirea cu ecou de la tastatură
Funcţia 02h	-	Afişarea unui caracter
Funcţia 03h	-	Intrarea auxiliară
Funcţia 04h	-	Ieşirea auxiliară
Funcţia 05h	-	Tipărirea unui caracter
Funcţia 06h	-	Intrare/ieşire directă fără ecou de la consolă
Funcţia 07h	-	Intrarea directă fără ecou de la consolă
Funcţia 08h	-	Citirea fără ecou da la tastatură
Funcţia 09h	-	Afişarea unui şir
Funcţia 0Ah	-	Intrarea prin zonă tampon de la tastatură
Funcţia 0Bh	-	Verificarea stării intrării
Funcţia 0Ch	-	Ştergerea zonei tampon de intrare şi citirea de la tastatură
Funcţia 0Dh	-	Resetarea discului
Funcţia 0Eh	-	Selectarea unităţii implicite
Funcţia 0Fh	-	Deschiderea unui fişier, utilizînd FCB
Funcţia 10h	-	Închiderea unui fişier, utilizînd FCB
Funcţia 11h	-	Căutarea primului fişier potrivit, utilizînd FCB
Funcţia 12h	-	Căutarea următorului fişier potrivit, utilizînd FCB
Funcţia 13h	-	Ştergerea unui fişier, utilizînd FCB
Funcţia 14h	-	Citirea secvenţială, utilizînd FCB
Funcţia 15h	-	Scrierea secvenţială, utilizînd FCB
Funcţia 16h	-	Crearea unui fişier, utilizînd FCB
Funcţia 17h	-	Redenumirea unui fişier, utilizînd FCB
Funcţia 18h	-	Rezervată
Funcţia 19h	-	Obţinerea unităţii curente implicite
Funcţia 1Ah	-	Modificarea adresei zonei de transfer a discului (DTA)
Funcţia 1Bh	-	Obţinerea informaţiilor FAT pentru unitatea curentă
Funcţia 1Ch	-	Obţinerea informaţiilor FAT pentru unitatea specificată
Funcţia 1Dh	-	Rezervată
Funcţia 1Eh	-	Rezervată
Funcţia 1Fh	-	Rezervată
Funcţia 20h	-	Rezervată
Funcţia 21h	-	Citirea în acces direct, utilizînd FCB

Funcţia 22h	-	Scrierea în acces direct, utilizînd FCB
Funcţia 23h	-	Obţinerea dimensiunii unui fişier, utilizînd FCB
Funcţia 24h	-	Poziţionarea cîmpului Articol în acces direct, utilizînd FCB
Funcţia 25h	-	Poziţionarea vectorului de întrerupere
Funcţia 26h	-	Crearea unui PSP
Funcţia 27h	-	Citirea multiplă în acces direct, utilizînd FCB
Funcţia 28h	-	Scrierea multiplă în acces direct, utilizînd FCB
Funcţia 29h	-	Analiza unui nume de fişier
Funcţia 2Ah	-	Obţinerea datei sistem curente
Funcţia 2Bh	-	Poziţionarea datei sistem curente
Funcţia 2Ch	-	Obţinerea orei sistem curente
Funcţia 2Dh	-	Poziţionarea orei sistem curente
Funcţia 2Eh	-	Poziţionarea/Repoziţionarea indicatorului VERIFY
Funcţia 2Fh	-	Obţinerea adresei DTA curente
Funcţia 30h	-	Obţinerea numărului versiunii DOS
Funcţia 31h	-	Păstrarea procesului
Funcţia 32h	-	Rezervată
Funcţia 33h	-	Poziţionarea/Obţinerea stării pentru Ctrl/C
Funcţia 34h	-	Rezervată
Funcţia 35h	-	Obţinerea vectorului de întrerupere
Funcţia 36h	-	Obţinerea spaţiului liber pe disc
Funcţia 37h	-	Rezervată
Funcţia 38h	-	Obţinerea/Poziţionarea informaţiilor dependente de ţară
Funcţia 39h	-	Crearea unui nou director (MKDIR)
Funcţia 3Ah	-	Ştergerea unui director (RMDIR)
Funcţia 3Bh	-	Schimbarea directorului (CHDIR)
Funcţia 3Ch	-	Crearea unui fişier (CREAT)
Funcţia 3Dh	-	Deschiderea unui fişier, utilizînd identificator logic
Funcţia 3Eh	-	Închiderea unui fişier, utilizînd identificator logic
Funcţia 3Fh	-	Citirea dintr-un fişier sau de la un dispozitiv, utilizînd identificator logic
Funcţia 40h	-	Scrierea într-un fişier sau la un dispozitiv, utilizînd identificator logic
Funcţia 41h	-	Ştergerea unui fişier (UNLINK)
Funcţia 42h	-	Poziţionarea pointerului de scriere/citire (LSEEK)

Funcţia 43h - Obţinerea/Poziţionarea atributelor fişierului (CHMOD)

Funcţia 44h - Control intrare/ieşire pentru dispozitive (IOCTL)

Funcţia 4400h - IOCTL: Obţinerea informaţiilor despre dispozitiv

Funcţia 4401h - IOCTL: Poziţionarea informaţiilor despre dispozitiv

Funcţia 4402h - IOCTL: Citirea de la un dispozitiv caracter

Funcţia 4403h - IOCTL: Scrierea la un dispozitiv caracter

Funcţia 4404h - IOCTL: Citirea de la un dispozitiv bloc

Funcţia 4405h - IOCTL: Srierea la un dispozitiv bloc

Funcţia 4406h - IOCTL: Obţinerea stării intrării

Funcţia 4407h - IOCTL: Obţinerea stării ieşirii

Funcţia 4408h - IOCTL: Dispozitivul se poate înlocui ?

Funcţia 4409h - IOCTL: Dispozitivul logic se află la distanţă ?

Funcţia 440Ah - IOCTL: Identificatorul logic se află la distanţă ?

Funcţia 440Bh - IOCTL: Modificarea numărului de reîncercări

Funcţia 440Ch - IOCTL: Cerere generică IOCTL pentru identificator logic

Funcţia 440Dh - IOCTL: Cerere generică IOCTL pentru dispozitive bloc

Funcţia 440Dh - Poziţionarea parametrilor dispozitivului (cod 40h)

Funcţia 440Dh - Returnarea parametrilor dispozitivului (cod 60h)

Funcţia 440Dh - Formatarea/Verificarea unei piste (cod 42h/62h) pe dispozitivul logic

Funcţia 440Eh - IOCTL: Obţinerea dispozitivului logic

Funcţia 440Fh - IOCTL: Selectarea dispozitivului logic

Funcţia 45h - Duplicarea identificatorului logic al unui fişier (DUP)

Funcţia 46h - Redirectarea unui identificator logic (FORCDUP)

Funcţia 47h - Obţinerea directorului curent

Funcţia 48h - Alocarea memoriei
Funcţia 49h - Eliberarea memoriei alocate
Funcţia 4Ah - Modificarea alocării memoriei (SET BLOCK)
Funcţia 4Bh - Încărcarea şi/sau execuţia unui program (EXEC)
Funcţia 4Ch - Terminarea procesului (EXIT)
Funcţia 4Dh - Obţinerea codului de retur al procesului fiu (WAIT)
Funcţia 4Eh - Găsirea primului fişier potrivit (FIND FIRST)
Funcţia 4Fh - Găsirea următorului fişier potrivit (FIND NEXT)
Funcţia 50h - Rezervată
Funcţia 51h - Rezervată
Funcţia 52h - Rezervată
Funcţia 53h - Rezervată
Funcţia 54h - Verificarea stării indicatorului VERIFY
Funcţia 55h - Rezervată
Funcţia 56h - Schimbarea intrării în director
Funcţia 57h - Obţinerea/Poziţionarea datei şi orei unui fişier
Funcţia 58h - Obţinerea/Selectarea strategiei de alocare
Funcţia 59h - Obţinerea informaţiilor suplimentare despre o eroare
Funcţia 5Ah - Crearea unui fişier temporar unic
Funcţia 5Bh - Crearea unui fişier nou
Funcţia 5Ch - Blocare/Deblocarea accesului la un fişier
Funcţia 5Dh - Rezervată
Funcţia 5Eh - Servicii de reţea
- Funcţia 5E00h - Obţinerea numelui maşinii
- Funcţia 5E02h - Poziţionarea antetului imprimantei de reţea
- Funcţia 5E03h - Obţinerea antetului imprimantei de reţea

Funcţia 5Fh - Redirectarea dispozitivelor de reţea
- Funcţia 5F02h - Obţinerea unei intrări în lista de redirectări
- Funcţia 5F03h - Redirectarea unui dispozitiv de reţea
- Funcţia 5F04h - Anularea redirectării unui dispozitiv de reţea

Funcţia 60h - Rezervată
Funcţia 61h - Rezervată
Funcţia 62h - Obţinerea adresei PSP-ului

1.2. Apelurile de funcţii DOS, în ordine alfabetică

Funcţia 02h - Afişarea unui caracter
Funcţia 09h - Afişarea unui şir
Funcţia 48h - Alocarea memoriei
Funcţia 29h - Analiza unui nume de fişier
Funcţia 5Ch - Blocare/Deblocarea accesului la un fişier
Funcţia 11h - Căutarea primului fişier potrivit, utilizînd FCB
Funcţia 12h - Căutarea următorului fişier potrivit, utilizînd FCB
Funcţia 21h - Citirea în acces direct, utilizînd FCB
Funcţia 01h - Citirea cu ecou de la tastatură
Funcţia 3Fh - Citirea dintr-un fişier sau de la un dispozitiv, utilizînd identificator logic
Funcţia 08h - Citirea fără ecou da la tastatură
Funcţia 27h - Citirea multiplă în acces direct, utilizînd FCB
Funcţia 14h - Citirea secvenţială, utilizînd FCB
Funcţia 44h - Control intrare/ieşire pentru dispozitive (IOCTL)

- Funcţia 4400h - IOCTL: Obţinerea informaţiilor despre dispozitiv
- Funcţia 4401h - IOCTL: Poziţionarea informaţiilor despre dispozitiv
- Funcţia 4402h - IOCTL: Citirea de la un dispozitiv caracter
- Funcţia 4403h - IOCTL: Scrierea la un dispozitiv caracter
- Funcţia 4404h - IOCTL: Citirea de la un dispozitiv bloc
- Funcţia 4405h - IOCTL: Srierea la un dispozitiv bloc
- Funcţia 4406h - IOCTL: Obţinerea stării intrării
- Funcţia 4407h - IOCTL: Obţinerea stării ieşirii
- Funcţia 4408h - IOCTL: Dispozitivul se poate înlocui ?
- Funcţia 4409h - IOCTL: Dispozitivul logic se află la distanţă ?
- Funcţia 440Ah - IOCTL: Identificatorul logic se află la distanţă ?
- Funcţia 440Bh - IOCTL: Modificarea numărului de reîncercări
- Funcţia 440Ch - IOCTL: Cerere generică IOCTL pentru identificator logic

Funcţia 440Dh - IOCTL: Cerere generică IOCTL pentru dispozitive bloc

  Funcţia 440Dh - Poziţionarea parametrilor dispozitivului (cod 40h)

  Funcţia 440Dh - Returnarea parametrilor dispozitivului (cod 60h)

  Funcţia 440Dh - Formatarea/Verificarea unei piste (cod 42h/62h) pe dispozitivul logic

Funcţia 440Eh - IOCTL: Obţinerea dispozitivului logic

Funcţia 440Fh - IOCTL: Selectarea dispozitivului logic

Funcţia 3Ch - Crearea unui fişier (CREAT)

Funcţia 5Bh - Crearea unui fişier nou

Funcţia 5Ah - Crearea unui fişier temporar unic

Funcţia 16h - Crearea unui fişier, utilizînd FCB

Funcţia 39h - Crearea unui nou director (MKDIR)

Funcţia 26h - Crearea unui PSP

Funcţia 0Fh - Deschiderea unui fişier, utilizînd FCB

Funcţia 3Dh - Deschiderea unui fişier, utilizînd identificator logic

Funcţia 45h - Duplicarea identificatorului logic al unui fişier (DUP)

Funcţia 49h - Eliberarea memoriei alocate

Funcţia 4Eh - Găsirea primului fişier potrivit (FIND FIRST)

Funcţia 4Fh - Găsirea următorului fişier potrivit (FIND NEXT)

Funcţia 04h - Ieşirea auxiliară

Funcţia 4Bh - Încărcarea şi/sau execuţia unui program (EXEC)

Funcţia 10h - Închiderea unui fişier, utilizînd FCB

Funcţia 3Eh - Închiderea unui fişier, utilizînd identificator logic

Funcţia 03h - Intrarea auxiliară

Funcţia 07h - Intrarea directă fără ecou de la consolă

Funcţia 0Ah - Intrarea prin zona tampon de la tastatură

Funcţia 06h - Intrare/ieşire directă fără ecou de la consolă

Funcţia 1Ah - Modificarea adresei zonei de transfer a discului (DTA)

Funcţia 4Ah - Modificarea alocării memoriei (SET BLOCK)

Funcţia 2Fh - Obţinerea adresei DTA curente

Funcţia 62h - Obţinerea adresei PSP-ului

Funcţia 4Dh - Obţinerea codului de retur al procesului fiu (WAIT)

- Funcţia 2Ah - Obţinerea datei sistem curente
- Funcţia 23h - Obţinerea dimensiunii unui fişier, utilizînd FCB
- Funcţia 47h - Obţinerea directorului curent
- Funcţia 1Bh - Obţinerea informaţiilor FAT pentru unitatea curentă
- Funcţia 1Ch - Obţinerea informaţiilor FAT pentru unitatea specificată
- Funcţia 59h - Obţinerea informaţiilor suplimentare despre o eroare
- Funcţia 30h - Obţinerea numărului versiunii DOS
- Funcţia 2Ch - Obţinerea orei sistem curente
- Funcţia 36h - Obţinerea spaţiului liber pe disc
- Funcţia 43h - Obţinerea/Poziţionarea atributelor fişierului (CHMOD)
- Funcţia 57h - Obţinerea/Poziţionarea datei şi orei unui fişier
- Funcţia 38h - Obţinerea/Poziţionarea informaţiilor dependente de ţară
- Funcţia 58h - Obţinerea/Selectarea strategiei de alocare
- Funcţia 19h - Obţinerea unităţii curente implicite
- Funcţia 35h - Obţinerea vectorului de întrerupere
- Funcţia 31h - Păstrarea procesului
- Funcţia 24h - Poziţionarea cîmpului Articol în acces direct, utilizînd FCB
- Funcţia 2Bh - Poziţionarea datei sistem curente
- Funcţia 2Dh - Poziţionarea orei sistem curente
- Funcţia 33h - Poziţionarea/Obţinerea stării pentru Ctrl/C
- Funcţia 42h - Poziţionarea pointerului de scriere/citire (LSEEK)
- Funcţia 2Eh - Poziţionarea/Repoziţionarea indicatorului VERIFY
- Funcţia 25h - Poziţionarea vectorului de întrerupere
- Funcţia 5Fh - Redirectarea dispozitivelor de reţea
    - Funcţia 5F02h - Obţinerea unei intrări în lista de redirectări
    - Funcţia 5F03h - Redirectarea unui dispozitiv de reţea
    - Funcţia 5F04h - Anularea redirectării unui dispozitiv de reţea
- Funcţia 17h - Redenumirea unui fişier, utilizînd FCB
- Funcţia 46h - Redirectarea unui identificator logic (FORCDUP)
- Funcţia 0Dh - Resetarea discului

Funcţia 3Bh - Schimbarea directorului (CHDIR)
Funcţia 56h - Schimbarea intrării în director
Funcţia 22h - Scrierea în acces direct, utilizînd FCB
Funcţia 28h - Scrierea multiplă în acces direct, utilizînd FCB
Funcţia 40h - Scrierea într-un fişier sau la un dispozitiv, utilizînd identificator logic
Funcţia 15h - Scrierea secvenţială, utilizînd FCB
Funcţia 0Eh - Selectarea unităţii implicite
Funcţia 5Eh - Servicii de reţea
Funcţia 5E00h - Obţinerea numelui maşinii
Funcţia 5E02h - Poziţionarea antetului imprimantei de reţea
Funcţia 5E03h - Obţinerea antetului imprimantei de reţea
Funcţia 0Ch - Ştergerea zonei tampon de intrare şi citirea de la tastatură
Funcţia 3Ah - Ştergerea unui director (RMDIR)
Funcţia 41h - Ştergerea unui fişier (UNLINK)
Funcţia 13h - Ştergerea unui fişier, utilizînd FCB
Funcţia 4Ch - Terminarea procesului (EXIT)
Funcţia 00h - Terminarea programului
Funcţia 05h - Tipărirea unui caracter
Funcţia 54h - Verificarea stării indicatorului VERIFY
Funcţia 0Bh - Verificarea stării intrării

1.3. Apelurile de funcţii DOS, pe categorii

1.3.1. Intrări/ieşiri pentru dispozitive caracter standard

Acestea tratează toate cererile de intrare/ieşire, de la şi la orice dispozitiv standard (consolă, imprimantă, porturi seriale). Dacă un program utilizează aceste funcţii, intrările şi ieşirile sale pot fi redirectate.

Funcţia 01h - Citirea cu ecou de la tastatură
Funcţia 02h - Afişarea unui caracter
Funcţia 03h - Intrarea auxiliară
Funcţia 04h - Ieşirea auxiliară
Funcţia 05h - Tipărirea unui caracter

Funcţia 06h - Intrare/ieşire directă fără ecou de la consolă
Funcţia 07h - Intrarea directă fără ecou de la consolă
Funcţia 08h - Citirea fără ecou da la tastatură
Funcţia 09h - Afişarea unui şir
Funcţia 0Ah - Intrarea prin zona tampon de la tastatură
Funcţia 0Bh - Verificarea stării intrării
Funcţia 0Ch - Ştergerea zonei tampon de intrare şi citirea de la tastatură

1.3.2. Gestiunea memoriei

DOS reţine care zone de memorie sînt alocate, scriind un bloc de control la începutul fiecărei zone de memorie. Acesta conţine dimensiunea zonei de memorie, numele procesului (dacă există) care o posedă şi un pointer la următoarea zonă de memorie.

Funcţia 48h - Alocarea memoriei
Funcţia 49h - Eliberarea memoriei alocate
Funcţia 4Ah - Modificarea alocării memoriei (SET BLOCK)
Funcţia 58h - Obţinerea/Selectarea strategiei de alocare

1.3.3. Gestiunea proceselor

DOS utilizează mai multe funcţii pentru a încărca, executa şi termina programele. Aplicaţiile care utilizează aceste funcţii pot gestiona alte programe.

Funcţia 31h - Păstrarea procesului
Funcţia 4Bh - Încărcarea şi/sau execuţia unui program (EXEC)
Funcţia 4Ch - Terminarea procesului (EXIT)
Funcţia 4Dh - Obţinerea codului de retur al procesului fiu (WAIT)
Funcţia 62h - Obţinerea adresei PSP-ului

1.3.4. Gestiunea fișierelor și directoarelor

1.3.4.1. Tratarea fișierelor prin FCB-uri

Pentru anumite apeluri de funcții, versiunile DOS mai vechi necesitau păstrarea unui FCB (cu diferite informații) pentru fiecare fișier. Aceste funcții care tratează fișierele prin FCB-uri au fost păstrate pentru compatibilitate.

Funcția 00h - Terminarea programului
Funcția 0Fh - Deschiderea unui fișier, utilizînd FCB
Funcția 10h - Închiderea unui fișier, utilizînd FCB
Funcția 11h - Căutarea primului fișier potrivit, utilizînd FCB
Funcția 12h - Căutarea următorului fișier potrivit, utilizînd FCB
Funcția 13h - Ștergerea unui fișier, utilizînd FCB
Funcția 14h - Citirea secvențială, utilizînd FCB
Funcția 15h - Scrierea secvențială, utilizînd FCB
Funcția 16h - Crearea unui fișier, utilizînd FCB
Funcția 17h - Redenumirea unui fișier, utilizînd FCB
Funcția 21h - Citirea în acces direct, utilizînd FCB
Funcția 22h - Scrierea în acces direct, utilizînd FCB
Funcția 23h - Obținerea dimensiunii unui fișier, utilizînd FCB
Funcția 24h - Poziționarea cîmpului Articol în acces direct, utilizînd FCB
Funcția 26h - Crearea unui PSP
Funcția 27h - Citirea multiplă în acces direct, utilizînd FCB
Funcția 28h - Scrierea multiplă în acces direct, utilizînd FCB

1.3.4.2. Tratarea fișierelor prin identificatoare logice

Noile versiuni DOS utilizează noțiunea de identificator logic, care se poate referi la un fișier, director sau dispozitiv. Există 5 identificatoare logice standard, deschise implicit.

Identificator logic de fişier	Dispozitiv standard	Observaţii
0	Intrare	Poate fi redirectat din linia de comandă.
1	Ieşire	Poate fi redirectat din linia de comandă.
2	Eroare	
3	Auxiliar	
4	Imprimantă	

DOS tratează un fişier ca un şir de octeţi, a cărui structură este stabilită de aplicaţie. Citirea şi scrierea necesită doar un pointer la o zonă de date şi un specificator pentru numărul de octeţi de scris sau citit.

Următoarele funcţii tratează fişierele utilizînd identificatoare logice:

Funcţia 3Ch - Crearea unui fişier (CREAT)
Funcţia 3Dh - Deschiderea unui fişier, utilizînd identificator logic
Funcţia 3Eh - Închiderea unui fişier, utilizînd identificator logic
Funcţia 3Fh - Citirea dintr-un fişier sau de la un dispozitiv, utilizînd identificator logic
Funcţia 40h - Scrierea într-un fişier sau la un dispozitiv, utilizînd identificator logic
Funcţia 42h - Poziţionarea pointerului de scriere/citire (LSEEK)
Funcţia 45h - Duplicarea identificatorului logic al unui fişier (DUP)
Funcţia 46h - Redirectarea unui identificator logic (FORCDUP)
Funcţia 5Ah - Crearea unui fişier temporar unic
Funcţia 5Bh - Crearea unui fişier nou

1.3.4.3. Partajarea fişierelor

Versiunea DOS 3.0 a introdus partajarea fişierelor, care permite ca mai multe procese să aibă acces la acelaşi fişier. Funcţiile care tratează această problemă au efect doar după execuţia comenzii DOS SHARE, care instalează partajarea fişierelor.

Funcția 3Dh - Deschiderea unui fișier, utilizînd identificator logic
Funcția 440hD - IOCTL: Modificarea numărului de reîncercări
Funcția 5Ch - Blocare/Deblocarea accesului la un fișier

1.3.4.4. Tratarea directoarelor

Versiunea DOS 2.0 a introdus directorul arborescent, cu adîncimea limitată doar de spațiul de stocare.

Tratarea directoarelor se face prin următoarele funcții:

Funcția 39h - Crearea unui nou director (MKDIR)
Funcția 3Ah - Ștergerea unui director (RMDIR)
Funcția 3Bh - Schimbarea directorului (CHDIR)
Funcția 41h - Ștergerea unui fișier (UNLINK)
Funcția 43h - Obținerea/Poziționarea atributelor fișierului (CHMOD)
Funcția 47h - Obținerea directorului curent
Funcția 4Eh - Găsirea primului fișier potrivit (FIND FIRST)
Funcția 4Fh - Găsirea următorului fișier potrivit (FIND NEXT)
Funcția 56h - Schimbarea intrării în director
Funcția 57h - Obținerea/Poziționarea datei și orei unui fișier

1.3.5. Gestiunea dispozitivelor

Funcția 44h conține mai multe subfuncții pentru operațiile cu dispozitive:

Funcția 4400h - IOCTL: Obținerea informațiilor despre dispozitiv
Funcția 4401h - IOCTL: Poziționarea informațiilor despre dispozitiv
Funcția 4402h - IOCTL: Citirea de la un dispozitiv caracter
Funcția 4403h - IOCTL: Scrierea la un dispozitiv caracter
Funcția 4404h - IOCTL: Citirea de la un dispozitiv bloc
Funcția 4405h - IOCTL: Srierea la un dispozitiv bloc
Funcția 4406h - IOCTL: Obținerea stării intrării
Funcția 4407h - IOCTL: Obținerea stării ieșirii

Funcția 4408h - IOCTL: Dispozitivul se poate înlocui ?
Funcția 440Ch - IOCTL: Cerere generică IOCTL pentru identificator logic
Funcția 440Dh - IOCTL: Cerere generică IOCTL pentru dispozitive bloc
Funcția 440Dh -Poziționarea parametrilor dispozitivului (cod 40h)
Funcția 440Dh -Returnarea parametrilor dispozitivului (cod 60h)
Funcția 440Dh -Formatarea/Verificarea unei piste pe (cod 42h/62h) dispozitivul logic
Funcția 440Eh - IOCTL: Obținerea dispozitivului logic
Funcția 440Fh - IOCTL: Selectarea dispozitivului logic

1.3.6. Rețele MicroSoft

Rețelele MicroSoft constau dintr-un server și una sau mai multe stații de lucru. DOS păstrează o lista care indică care drivere și dispozitive ale unei stații de lucru au fost redirectate către server.

Aceste rețele sînt tratate de următoarele funcții:

Funcția 440Dh - IOCTL: Dispozitivul logic se află la distanță ?
Funcția 440Dh - IOCTL: Identificatorul logic se află la distanță ?
Funcția 5E00h - Obținerea numelui mașinii
Funcția 5E02h - Poziționarea antetului imprimantei de rețea
Funcția 5F02h - Obținerea unei intrări în lista de redirectări
Funcția 5F03h - Redirectarea unui dispozitiv de rețea
Funcția 5F04h - Anularea redirectării unui dispozitiv de rețea

1.3.7. Diferite operații sistem

Restul funcțiilor sistem tratează resurse diverse, cum ar fi drivere, adrese și ceas.

Funcția 0Dh - Resetarea discului
Funcția 0Eh - Selectarea unității implicite
Funcția 19h - Obținerea unității curente implicite

Funcţia 1Ah	- Modificarea adresei zonei de transfer a discului (DTA)
Funcţia 1Bh	- Obţinerea informaţiilor FAT pentru unitatea curentă
Funcţia 1Ch	- Obţinerea informaţiilor FAT pentru unitatea specificată
Funcţia 25h	- Poziţionarea vectorului de întrerupere
Funcţia 29h	- Analiza unui nume de fişier
Funcţia 2Ah	- Obţinerea datei sistem curente
Funcţia 2Bh	- Poziţionarea datei sistem curente
Funcţia 2Ch	- Obţinerea orei sistem curente
Funcţia 2Dh	- Poziţionarea orei sistem curente
Funcţia 2Eh	- Poziţionarea/Repoziţionarea indicatorului VERIFY
Funcţia 2Fh	- Obţinerea adresei DTA curente
Funcţia 30h	- Obţinerea numărului versiunii DOS
Funcţia 33h	- Poziţionarea/Obţinerea stării pentru Ctrl/C
Funcţia 35h	- Obţinerea vectorului de întrerupere
Funcţia 36h	- Obţinerea spaţiului liber pe disc
Funcţia 38h	- Obţinerea/Poziţionarea informaţiilor dependente de ţară
Funcţia 54h	- Verificarea stării indicatorului VERIFY
Funcţia 59h	- Obţinerea informaţiilor suplimentare despre o eroare

1.4. Apeluri sistem vechi

Acestea au fost păstrate pentru compatibilitate cu versiunile DOS mai vechi. Toate au însă corespondenţe noi, şi se recomandă a nu fi utilizate dacă nu este strict necesar.

Apel vechi	Corespondent nou
00h	4Ch
0Fh	40h
10h	3Eh
11h	4Eh
12h	4Fh
13h	41h
14h	3Fh

15h	40h
16h	3Ch
	5Ah
	5Bh
17h	56h
21h	3Fh
22h	40h
23h	42h
24h	42h
26h	4B00h
27h	42h
	3Fh
28h	42h
	40h

2. Descrierea detaliată a apelurilor sistem

2.1. Funcția 00h - Terminarea programului

Funcția 00h realizează terminarea unui program și revenirea în DOS. Controlul este trecut vectorului terminator din PSP (se iese în procesul părinte). Vectorii de întrerupere DOS 22h-24h (Terminarea programului, Ctrl/Break, Eroare critică) sînt repoziționați la valorile salvate din PSP-ul procesului părinte. Sînt golite toate zonele tampon de fișiere.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 00h;
CS - încărcat cu adresa de segment a PSP-ului programului de terminat.

Funcția nu returnează nimic.

Observații:

1. Funcția este echivalentă cu INT 20h.

2. În noile versiuni DOS, este de preferat utilizarea funcției 4Ch.

3. Pentru informații suplimentare, a se vedea funcțiile 4Ch, 31h și INT 20h, INT 27h.

ATENȚIE!

Înainte de apel, toate fișierele trebuie să fie închise, altfel lungimea celor modificate nu va fi înregistrată corect în director.

2.2. Funcţia 01h - Citirea cu ecou de la tastatură

Funcţia 01h realizează citirea unui caracter de la dispozitivul de intrare standard (în mod normal tastatura) şi afişarea lui la dispozitivul standard de ieşire (în mod normal ecranul). Dacă se detectează Ctrl/Break sau Ctrl/C, se generează INT 23h.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 01h.

Funcţia returnează în registrul AL caracterul citit.

Observaţii:

1. Pentru tastele speciale (cursor, taste funcţionale F1-F12, ...) sînt necesare două apeluri. Primul returnează 0 în AL, al doilea codul ASCII extins.

2. Pentru informaţii suplimentare, a se vedea funcţiile 06h, 07h, 08h, 0Ah, 0Ch.

2.3. Funcţia 02h - Afişarea unui caracter

Funcţia 02h trimite la dispozitivul de ieşire standard caracterul a cărui cod ASCII se află în registrul DL. Caracterul 'backspace' (ASCII 8) este tratat prin deplasarea cursorului cu o poziţie la stînga. Dacă se detectează Ctrl/Break sau Ctrl/C, se generează INT 23h, după afişarea caracterului.

Pentru apelul funcţiei, se folosesc regiştrii:

AH-încărcat cu 02h;
DL-încărcat cu codul caracterului.

Funcţia nu returnează nimic.

Pentru informaţii suplimentare, a se vedea funcţiile 06h, 09h.



2.4. Funcția 03h - Intrarea auxiliară

Funcția 03h citește (așteaptă) un caracter de la dispozitivul auxiliar standard (AUX, în mod normal prima interfață serială COM1). La pornirea sistemului, DOS inițializează implicit portul COM1 astfel:

- 2400 bauds;
- fără paritate;
- un bit de stop;
- 8 biți de date.

Dacă se detectează Ctrl/Break sau Ctrl/C, se generează INT 23h.

Pentru apelul funcției, se folosește registrul:

AH-încărcat cu 03h.

Funcția returnează în registrul AL caracterul citit.

Observații:

1. Intrarea nu este salvată în zona tampon și nici supravegheată prin întreruperi, astfel încît caracterul trebuie prelucrat rapid, pentru a nu se pierde.

2. Funcția nu returnează starea dispozitivului auxiliar și nici cod de eroare.

3. Pentru informații suplimentare, a se vedea funcțiile 03h, 3Fh și INT 14h.

2.5. Funcția 04h - Ieșirea auxiliară

Funcția 04h trimite caracterul aflat în registrul DL la dispozitivul auxiliar standard AUX (implicit COM1). Dacă se detectează Ctrl/Break sau Ctrl/C, se generează INT 23h.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 04h;
DL - încărcat cu codul caracterului.

Funcţia nu returnează nimic.

Observaţii:

1. Dacă dispozitivul este ocupat, se aşteaptă pînă cînd el devine operaţional.

2. Funcţia nu returnează starea dispozitivului auxiliar şi nici cod de eroare.

3. Pentru informaţii suplimentare, a se vedea funcţiile 03h, 40h şi INT 14h.

2.6. Funcţia 05h - Tipărirea unui caracter

Funcţia 05h trimite caracterul al cărui cod ASCII se află în registrul DL la dispozitivul standard de imprimare (în mod normal, prima interfaţă paralelă LPT1). Dacă se detectează Ctrl/Break sau Ctrl/C, se generează INT 23h.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 05h;
DL - încărcat cu codul caracterului.

Funcţia nu returnează nimic.

Observaţii:

1. Dacă dispozitivul este ocupat, se aşteaptă pînă cînd el devine operaţional.

2. Întrucît funcţia nu returnează starea dispozitivului şi nici cod de eroare, este preferabil să se deschidă LPT1 ca şi fişier.

3. Ieșirea poate fi redirectată la un port serial prin comanda DOS MODE.

4. Pentru informații suplimentare, a se vedea funcția 40h și INT 17h.

2.7. Funcția 06h - Intrare/ieșire directă fără ecou de la consolă

Funcția 06h preia un caracter de la dispozitivul de intrare standard (dacă există un caracter pregătit) sau trimite un caracter la dispozitivul de ieșire standard. Nu se face ecou și nu se verifică Ctrl/Break sau Ctrl/C.

Pentru apelul funcției, se folosesc regiștrii:

- la citirea unui caracter:

AH - încărcat cu 06h;
DL - încărcat cu FFh.

Funcția returnează în indicatorul ZF valoarea 1 dacă nu există caracter pregătit, respectiv 0 în ZF și caracterul citit în registrul AL în caz contrar.

- la scrierea unui caracter:

AH - încărcat cu 06h;
DL - încărcat cu codul caracterului.

Funcția nu returnează nimic.

Observații

1. Valoarea 0 returnată în registrul AL la citire indică prezența unui caracter special la intrare (cursor, cheie funcțională, etc). Un al doilea apel va returna codul extins.

2. Pentru informații suplimentare, a se vedea funcțiile 01h, 02h, 07h, 08h, 09h, 0Ah, 0Bh, 0Ch.

2.8. Funcţia 07h - Intrarea directă fără ecou de la consolă

Funcţia 07h citeşte (aşteaptă) un caracter de la dispozitivul de intrare standard şi îl returnează în registrul AL. Nu se face ecou şi nu se verifică Ctrl/Break sau Ctrl/C.

Pentru apelul funcţiei, se foloseşte registrul:

AH - încărcat cu 07h.

Funcţia returnează în registrul AL caracterul citit.

Observaţii:

1. Valoarea 0 returnată în registrul AL la citire indică prezenţa unui caracter special la intrare (cursor, cheie funcţională, etc). Un al doilea apel va returna codul extins.
2. Pentru informaţii suplimentare, a se vedea 08h, 0Ah, 0Bh, 0Ch.

2.9. Funcţia 08h - Citirea fără ecou da la tastatură

Funcţia 08h citeşte (aşteaptă) un caracter de la dispozitivul de intrare standard şi îl returnează în registrul AL. Nu se face ecou. Dacă se detectează Ctrl/Break sau Ctrl/C, se generează INT 23h.

Pentru apelul funcţiei, se foloseşte registrul:

AH - încărcat cu 08h.

Funcţia returnează în AL caracterul citit.

Observaţii:

1. Valoarea 0 returnată în registrul AL la citire indică prezenţa unui caracter special la intrare (cursor, cheie funcţională, etc). Un al doilea apel va returna codul extins.



2. Pentru informaţii suplimentare, a se vedea funcţiile 01h, 06h, 07h, 0Ah, 0Bh, 0Ch.

2.10. Funcţia 09h - Afişarea unui şir

Funcţia 09h trimite la dispozitivul de ieşire standard un şir de caractere terminat prin '$' (caracterul '$' nu este tipărit), al cărui deplasament faţă de adresa segment din registrul DS se află în registrul DX. Caracterul backspace este tratat ca la funcţia 02h. Dacă se detectează Ctrl/Break sau Ctrl/C, se generează INT 23h.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 09h;
DS:DX- încărcat cu un pointer la şir.

Funcţia nu returnează nimic.

Observaţii:

1. Sînt tipărite toate caracterele, în afară de '$' şi de caracterele speciale.

2. Caracterul '$' se poate tipări cu funcţia 40h.

3. Afişarea caracterului 'linie nouă' se face în mod normal prin perechea CR + LF (ASCII 13h urmat de ASCII 0Ah).

4. Pentru informaţii suplimentare, a se vedea funcţiile 02h, 06h.

2.11. Funcţia 0Ah - Intrarea prin zonă tampon de la tastatură

Funcţia 0Ah citeşte cu ecou un şir de caractere de la dispozitivul de intrare standard, pînă la întîlnirea caracterului CR. Şirul (inclusiv CR) este memorat de la deplasamentul 2 (octetul 3) al zonei tampon al cărui deplasament faţă de adresa segment din registrul DS se află în registrul DX. Dacă se detectează Ctrl/Break sau Ctrl/C, se generează INT 23h.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 0Ah;
DS:DX - încărcat cu un pointer la zona tampon de intrare.

Observaţii:

1. La apel, primul octet al zonei tampon trebuie să conţină lungimea maximă a şirului, inclusiv CR (1 - 255).

2. La returnare, primul octet nu se modifică, al doilea este poziţionat la lungimea actuală a şirului (exclusiv CR), iar de la octetul al treilea se află şirul citit.

3. Dacă în timpul citirii se atinge numărul maxim de caractere - 1, se trimite 07h (clopoţel) la dispozitivul de ieşire standard pentru fiecare caracter în plus, acestea fiind ignorate pîna la CR (exclusiv).

4. Comenzile DOS de editare sînt active în timpul citirii.

5. Pentru informaţii suplimentare, a se vedea funcţiile 01h, 06h, 07h, 08h, 0Ch.

2.12. Funcţia 0Bh - Verificarea stării intrării

Funcţia 0Bh verifică dacă există sau nu un caracter pregătit la dispozitivul de intrare standard. Dacă se detectează Ctrl/Break sau Ctrl/C, se generează INT 23h.

Pentru apelul funcţiei, se foloseşte registrul:

AH - încărcat cu 0Bh.

Funcţia returnează în registrul AL valoarea FFh dacă există un caracter pregătit la dispozitivul de intrare standard, respectiv 0 în caz contrar.

Observații:

1. Este o metodă simplă de a verifica apariția unui Ctrl/Break în timpul unor prelucrări de durată mare.

2. Se recomandă utilizarea acestei funcții înaintea apelului funcțiilor 01H, 07h și 08h.

3. Pentru informații suplimentare, a se vedea INT 16h și funcțiile 01h, 06h, 07h, 08h.

2.13. Funcția 0Ch - Ștergerea zonei tampon de intrare și citirea de la tastatură

Funcția 0Ch șterge toate caracterele din zona tampon de intrare și apelează funcția DOS specificată în registrul AL. Sistemul este forțat să aștepte tastarea unui caracter.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 0Ch.
AL - încărcat cu codul funcției de apelat (01h, 06h, 07h, 08h sau 0Ah). Orice altă valoare realizează doar ștergerea zonei tampon de intrare.

Funcția returnează în AL caracterul citit (în afara cazului în care s-a apelat funcția 0Ah), respectiv valoarea 0 dacă s-a făcut doar ștergerea zonei tampon, fără prelucrări ulterioare.

Observații

1. Dacă se dorește doar ștergerea zonei tampon, funcția poate fi apelată cu AL = 06h și DL = FFh, verificîndu-se astfel dacă există caractere de citit la dispozitivul de intrare standard, fă·ă ca acestea sa fie citite.

2. Pentru informații suplimentare, a se vedea funcțiile 01h, 06h, 07h, 08h, 0Ah.

2.14. Funcţia 0Dh - Resetarea discului

Funcţia 0Dh goleşte toate zonele tampon de fişiere, fără a închide fişierele. Toate zonele tampon modificate sînt scrise, şi toate zonele tampon ale memoriei cache interne sînt marcate ca fiind libere. Funcţia poate fi folosită pentru tratarea lui Ctrl/Break.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 0Dh.

Funcţia nu returnează nimic.

Observaţie:

Pentru a avea certitudinea că în director este înregistrată lungimea corectă a fişierelor modificate, se recomandă închiderea lor folosind una din funcţiile 10h sau 3Eh.

2.15. Funcţia 0Eh - Selectarea unităţii implicite

Funcţia 0Eh selectează unitatea specificată (0 = A, 1 = B, etc) ca fiind unitatea logică curentă.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 0EH;
DL - încărcat cu numărul unităţii logice.

Funcţia returnează în AL numărul de unităţi logice din sistem.

Observaţii:

1. Pentru test de eroare, se poate utiliza funcţia 19h.

2. Pe sisteme cu o singură unitate de disc flexibil, acesta este înregistrat ca reprezentînd două unităţi logice (A: şi B:).

3. Valoarea returnată în AL conţine unităţile de toate tipurile, inclusiv discurile Winchester şi unităţile logice.

4. Pentru determinarea numărului de unităţi fizice de disc flexibil, se poate utiliza INT 11h, iar pentru discuri Winchester, INT 13h.

5. Versiunea DOS 3.0 şi următoarele returnează în AL valoarea LASTDRIVE specificată în fişierul CONFIG.SYS (implicit 5).

6. Valoarea returnată în AL trebuie tratată cu atenţie. De exemplu, dacă se returnează 5, aceasta nu înseamnă că A, B, C, D, E desemnează unităţi valide.

2.16. Funcţia 0Fh - Deschiderea unui fişier, utilizînd FCB

Funcţia 0Fh deschide în mod acces compatibilitate un fişier existent, utilizînd un bloc de control al fişierului (FCB), cu deplasamentul faţă de adresa segment din DS specificat în registrul DX.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 0Fh;
DS:DX - încărcat cu un pointer la un FCB nedeschis.

Funcţia returnează în registrul AL valoarea 0 dacă s-a găsit o intrare în director pentru fişierul specificat, respectiv FFh în caz contrar.

FCB-ul este inițializat după cum urmează:

Depla-sament	Lungime	Cîmp	Inițializat la	Inițializat de
00h	1	Cod unitate	Unitate specificată	Utilizator
01h	8	Nume fișier	Nume fișier	Utilizator
09h	3	Extensie	Extensie	Utilizator
0Ch	2	Bloc curent	00h	DOS
0Eh	2	Dimensiune articol	80h	DOS
10h	4	Dimensiune fișier	Valoare din director (octeți)	DOS
14h	2	Data	Valoarea din director	DOS
16h	2	Ora	Valoarea din director	DOS
18h	2	Rezervat		
20h	1	Articol curent		Utilizator
21h	4	Articol în acces direct		Utilizator

Pentru fișiere cu atribute speciale, se folosește un FCB extins, avînd 3 cîmpuri suplimentare, începînd de la -O7h.

Depla-sament	Lungime	Cîmp	Inițializat la	Inițializat de
-07h	1	Identificator FCB extins	FFh	Utilizator
-06h	5	Rezervat	Zerouri	Utilizator
-01h	1	Octet de atribut	Atribut fișier	Utilizator

Observații:

1. Un FCB nedeschis conține primii 12 octeți (Cod unitate, Nume fișier și Extensie) plus suficient spațiu rezervat pentru celelalte cîmpuri. Un FCB deschis are și restul cîmpurilor completate.

2. Numele și extensia fișierului se aliniază la stînga și se completează cu spații la dreapta, pînă la dimensiunea specificată.

3. Un bloc reprezintă un grup de 128 articole. Cîmpurile Bloc curent și Articol curent formează pointerul la articol.

4. Valoarea orei este împachetată pe 2 octeți, în forma:

Bit	15 14 13 12 11	10 9 8 7 6 5	4 3 2 1 0
Semnificație	Oră	Minut	Secundă/2

Extragerea se face cu măști:

Secundă/2 (0 - 30) = (ora & 001Fh)
Minut (0 - 59) = ((ora & 07E0h) >> 5)
Ora (0 - 23) = ((ora & 0F800h)) >> 11)

5. Valoarea datei este împachetată pe 2 octeți, în forma:

Bit	15 14 13 12 11 10 9	8 7 6 5	4 3 2 1 0
Semnificație	An	Luna	Zi

Extragerea se face cu măști:

Zi (0 - 31) = (data &001Fh)
Luna (1 - 12) = ((data & 01E0h) >> 5)
An (1 - 119) = ((data & F800h) >> 9)

Anul este numărat începînd cu 1981, deci pentru valoara reală se adună 1980 (07BCh).

6. Dacă codul unității a fost 0, el este schimbat automat cu unitatea implicită curentă (A = 1, B = 2, etc).

7. Dacă se doreşte schimbarea valorii pentru dimensiunea articolului, aceasta se va face după deschidere, înaintea oricărei alte operaţii cu discul.

8. Cîmpul Articol curent este utilizat la operaţii de scriere/citire secvenţiale şi trebuie poziţionat la numărul de articole (0 - 127) care se doresc a fi accesate în interiorul blocului curent, înaintea oricăror operaţii de citire/scriere secvenţială.

9. Cîmpul Articol în acces direct este utilizat la operaţii de citire/scriere în acces direct şi va fi poziţionat înaintea oricărei operaţii de citire/scriere la numărul de articole relativ la începutul fişierului de accesat (începînd cu valoarea 0). Octetul al patrulea din acest cîmp este folosit doar dacă dimensiunea articolului este mai mică de 64. Acest cîmp trebuie poziţionat înaintea oricarei operaţii de scriere/citire în acces direct.

10. În noile versiuni DOS, este de preferat utilizarea funcţiei 40h.

11. Pentru informaţii suplimentare, a se vedea funcţiile 3Dh, 10h, 29h şi FCB.

2.17. Funcţia 10h - Închiderea unui fişier, utilizînd FCB

Funcţia 10h închide un fişier, utilizînd un bloc de control al fişierului, cu deplasamentul faţă de adresa segment din DS specificat în registrul DX.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 10h;
DS:DX - încărcat cu un pointer la un FCB deschis.

Funcţia returnează în registrul AL valoarea 0 dacă s-a găsit o intrare în director pentru fişierul specificat, respectiv FFh în caz contrar.

Observații:

1. Orice fișier, chiar nemodificat, se recomandă a fi închis, dacă nu mai este accesat.

2. În noile versiuni, este de preferat utilizarea funcției 3Eh.

3. Pentru informații suplimentare, a se vedea funcțiile 0Fh, 3Eh și FCB.

ATENȚIE!

Încercarea de a inchide un fișier inexistent poate avea efecte nedorite.

2.18. Funcția 11h - Căutarea primului fișier potrivit, utilizînd FCB

Funcția 11h caută în directorul curent primul fișier al cărui nume se potrivește cu cel specificat în blocul de control avînd deplasamentul față de adresa segment din DS în registrul DX. Numele de fișier din FCB poate conține caracterul '?' sau '*'. Pentru fișiere ascunse sau sistem, DX trebuie să pointeze la primul octet al unui FCB extins.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 11h;
DS:DX - încărcat cu un pointer la un FCB nedeschis.

Funcția returnează în registrul AL valoarea 0 dacă s-a găsit o intrare în director pentru fișierul specificat, respectiv FFh în caz contrar.

Dacă s-a găsit intrarea în director pentru numele de fișier din FCB, se creează un FCB nedeschis de același tip la adresa DTA curentă, astfel:

- Pentru FCB normal, primul octet din DTA este poziționat la numărul unității specificate în căutare (1 = A, 2 = B, ...), iar următorii 32 de octeți conțin intrarea în director.

- Pentru FCB extins, primul octet din DTA este poziţionat la FFh, următorii 5 la 00h, iar următorul la valoarea octetului de atribut din FCB-ul căutat. Cei 33 de octeţi rămaşi sînt poziţionaţi ca la un FCB normal.

Observaţii:

1. Cîmpul de atribut al FCB-ului extins permite căutarea diferitelor tipuri de fişiere sau combinaţii ale acestora:

   00h - fişiere normale;
   08h - eticheta volumului;
   02h - fişiere ascunse;
   04h - fişiere sistem;
   10h - intrări în director.

Ultimele trei şi orice combinaţie a lor caută şi toate fişierele normale.

2. Atributele arhivă (20h) şi "read-only" (01h) nu se pot folosi ca şi criterii de căutare.

3. Funcţia 29h se poate utiliza pentru crearea unui FCB nedeschis, pregătind căutarea.

4. Înainte de apel, se recomandă a se asigura că există spaţiu suficient pentru FCB în DTA. La nevoie, se poate utiliza funcţia 1Ah pentru modificarea adreşei DTA.

5. Dacă se continuă căutarea prin funcţia 12h, FCB-ul "găsit" trebuie salvat în altă parte, înaintea efectuării oricaror operaţii de intrare/ieşire asupra lui.

6. În noile versiuni, este de preferat utilizarea funcţiei 4Eh.

4. Pentru informaţii suplimentare, a se vedea funcţiile 12h, 1Ah, 29h, 4Eh, 4Fh, FCB şi PSP.

2.19. Funcţia 12h - Căutarea următorului fişier potrivit, utilizînd FCB

Funcţia 12h se utilizează în căutarea următorului fişier cu un nume care se potriveşte cu cel specificat în blocul de control avînd deplasamentul faţă de adresa segment din DS în registrul DX. Numele de fişier din FCB poate conţine caracterul '?'sau '*'. Pentru fişiere ascunse sau sistem, DX trebuie să pointeze la primul octet al unui FCB extins.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 12h;
DS:DX - încărcat cu un pointer la un FCB nedeschis.

Funcţia returnează în registrul AL valoarea 0 dacă s-a găsit o intrare în director pentru fişierul specificat, respectiv FFh în caz contrar.

Observaţii:

1. Se recomandă utilizarea funcţiei 12h după un apel al funcţiei 11h sau înaintea unui alt apel al funcţiei 12h. Apelurile următoare vor completa DTA cu noua intrare în director.

2. Dacă se continuă căutarea prin funcţia 12h, FCB-ul "găsit" trebuie salvat în altă parte, înaintea efectuării oricăror operaţii de intrare/ieşire asupra lui.

3. În noile versiuni, este de preferat utilizarea funcţiei 4Fh.

4. Pentru informaţii suplimentare, a se vedea funcţiile 11h, 4Eh, 4Fh şi FCB.

2.20. Funcţia 13h - Ştergerea unui fişier, utilizînd FCB

Funcţia 13h şterge un fişier, utilizînd un bloc de control cu deplasamentul faţă de adresa segment din DS specificat în registrul DX. Numele de fişier poate conţine caracterul '?', toate fişierele cu nume potrivite fiind şterse.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 13h;
DS:DX - încărcat cu un pointer la un FCB nedeschis.

Funcţia returnează în registrul AL valoarea 0 dacă s-a găsit o intrare în director pentru fişierul specificat, respectiv FFh în caz contrar.

Observaţii:

1. Fişierele "read-only" nu vor fi şterse.

2. În noile versiuni, este de preferat utilizarea funcţiei 41h.

ATENŢIE!

Fişierele trebuie închise, înainte de a fi şterse.

2.21. Funcţia 14h - Citirea secvenţială, utilizînd FCB

Funcţia 14h citeşte un articol din fişier (în DTA), utilizînd un bloc de control cu deplasamentul faţă de adresa segment din DS specificat în registrul DX. Articolul pointat de cîmpurile Bloc curent (deplasament 0Ch din FCB) şi Articol curent (deplasament 20h în FCB) este încărcat în DTA, după care cele două cîmpuri sînt actualizate (incrementate). Lungimea articolului se ia din cîmpul Dimensiune articol (deplasament 0Eh în FCB).

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 14h;
DS:DX- încărcat cu un pointer la un FCB deschis.

Funcţia returnează în registrul AL următoarele:

00h - citire cu succes;
01h - s-a găsit EOF, fără a se citi date;
02h - DTA prea mic, se anulează citirea;

03h - s-a găsit EOF înaintea citirii unui articol întreg (se face completare cu zerouri).

Observaţii:

1. Este recomandată poziţionarea cîmpului Articol curent la 0 înaintea unei citiri secvenţiale (funcţia 0Fh nu îl iniţializează).

2. În noile versiuni, este de preferat utilizarea funcţiei 3Fh.

3. Pentru informaţii suplimentare, a se vedea funcţiile 0Fh, 1Ah, 21h, 27h, 3Fh şi FCB.

2.22. Funcţia 15h - Scrierea secvenţială, utilizînd FCB

Funcţia 15h scrie un articol (din DTA) în fişier, utilizînd un bloc de control cu deplasamentul faţă de adresa segment din DS specificat în registrul DX. Articolul pointat de cîmpurile Bloc curent (deplasament 0Ch din FCB) şi Articol curent (deplasament 20h în FCB) este încărcat din DTA, după care cele două cîmpuri sînt incrementate. Lungimea articolului se ia din cîmpul Dimensiune articol (deplasament 0Eh în FCB).

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 14h;
DS:DX - încărcat cu un pointer la un FCB deschis.

Funcţia returnează în registrul AL următoarele:

00h - scriere cu succes;
01h - disc plin, se anulează scrierea;
02h - DTA prea mic, se anulează scrierea.

Observaţii:

1. Este recomandată poziţionarea cîmpului Articol curent la 0 înaintea unei scrieri secvenţiale (funcţia 0Fh nu îl iniţializează).

2. Dacă cîmpul Dimensiune articol este mai mic decît un sector, datele sînt stocate într-o zonă tampon, fiind scrise pe disc cînd se atinge dimensiunea unui sector, sau cînd se închide fişierul sau se face un apel al funcţiei 0Dh.

3. În fişierele "read-only" nu se va scrie.

4. În noile versiuni, este de preferat utilizarea funcţiei 40h.

5. Pentru informaţii suplimentare, a se vedea funcţiile 0Fh, 1Ah, 22h, 28h, 40h şi FCB.

2.23. Funcţia 16h - Crearea unui fişier, utilizînd FCB

Funcţia 16h creează un fişier, utilizînd un FCB a cărui deplasament faţă de adresa segment din registrul DS se află în registrul DX. FCB-ul este completat la fel ca la funcţia 0Fh.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 16h;
DS:DX - încărcat cu un pointer la un FCB nedeschis.

Funcţia returnează în registrul AL valoarea 00h dacă fişierul a fost creat, respectiv FFh în caz contrar (nu există intrare vidă în director).

Observaţii:

1. Fişierul este creat dacă s-a găsit o intrare în director potrivită cu numele de fişier specificat, sau dacă s-a găsit o intrare vidă în director. În ambele cazuri, lungimea fişierului este poziţionată la 0 (dacă fişierul există, el va fi şters).

2. Fişierului i se poate asigna un atribut, utilizînd un FCB extins.

3. În noile versiuni, este de preferat utilizarea uneia din funcţiile 3Ch, 5Ah sau 5Bh.

4. Pentru informaţii suplimentare, a se vedea funcţiile 0Fh, 3Ch, 5Bh şi FCB.

2.24. Funcţia 17h - Redenumirea unui fişier, utilizînd FCB

Funcţia 17h schimbă numele unui fişier existent, utilizînd un FCB (a cărui deplasament faţă de adresa segment din registrul DS se află în registrul DX) care conţine numărul unităţii, vechiul nume şi extensie a fişierului, urmate de la deplasamentul 11h de noul nume şi de noua extensie.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 17h;
DS:DX - încărcat cu un pointer la un FCB modificat.

Funcţia returnează în registrul AL valoarea 00h dacă operaţia a reuşit, respectiv FFh în caz contrar (nu există intrare vidă în director sau fişierul există deja).

Observaţii:

1. Ambele nume de fişier pot conţine caracterul '?'. La primul nume, el are semnificaţia obişnuită, la al doilea el specifică faptul că nu se va schimba caracterul din poziţia respectivă.

2. Ca şi nume nou, nu se poate utiliza numele unui fişier existent.

3. Nu se pot redenumi fişierele ascunse, sistem, subdirectoare sau "read-only".

4. În noile versiuni, este de preferat utilizarea funcţiei 56h.

2.25. Funcţia 18h - Rezervată

2.26. Funcţia 19h - Obţinerea unităţii curente implicite

Funcţia 19h returnează în registrul AL numărul unităţii curente implicite.

Pentru apelul funcţiei, se foloseşte registrul:

AH - încărcat cu 19h.

Funcţia returnează în AL numărul unităţii curente.

Observaţie:

Numărul unităţii returnat în AL este mai mic cu 1 decît la majoritatea celorlalte funcţii (0 = A, 1 = B, etc).

2.27. Funcţia 1Ah- Modificarea adresei zonei de transfer a discului (DTA)

Funcţia 1Ah modifică adresa zonei de transfer a discului, la valoarea avînd deplasamentul faţă de adresa segment din registrul DS în registrul DX. Implicit, în momentul lansării unui program, pentru DTA sînt folosiţi 128 octeţi, de la deplasamentul 80h din PSP.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 19h;
DS:DX - încărcat cu adresa DTA.

Funcţia nu returnează nimic.

Observaţii:

1. Trebuie rezervat spaţiu suficient pentru DTA, întrucît nu se permit transferuri care depăşesc graniţele acestei zone.

2. Toate accesele la fişiere utilizînd FCB se fac prin DTA, iar funcţiile de căutare a fişierelor (11h, 12h, 4Eh, 4Fh) îşi plasează datele în DTA.

3. Se recomandă utilizarea împreună cu funcţiile de scriere, citire şi căutare fişiere.

4. DTA este globală, astfel încît modificarea adresei ei în proceduri recursive sau reentrante trebuie făcută cu atenţie.

3. Pentru informaţii suplimentare, a se vedea funcţia 2Fh şi DTA.

2.28. Funcţia 1Bh - Obţinerea informaţiilor FAT pentru unitatea curentă

Funcţia 1Bh returnează informaţii despre dimensiunea şi tipul unităţii implicite.

Pentru apelul funcţiei, se foloseşte registrul:

AH-încărcat cu 1Bh.

Funcţia returnează următoarele:

- în AL -numărul de sectoare pe unitate de alocare a discului (cluster);
- în CX - numărul de octeţi pe sector;
- în DX -numărul de unităţi de alocare ale discului pe unitate;
- în DS:BX - pointer la octetul de identificare FAT (acesta redă tipul unităţii).

Observaţii:

1. Pentru informaţii mai sigure despre tipul unităţii, se recomandă utilizarea INT 25h pentru citirea Blocului Parametrilor BIOS (BPB) din înregistrarea de iniţializare (sectorul de boot).

2. Pentru informaţii suplimentare, a se vedea funcţiile 1Ch, 32h, 36h şi FAT.

ATENŢIE!

Apelul acestei funcţii distruge registrul DS.

2.29. Funcţia 1Ch-Obţinerea informaţiilor FAT pentru unitatea specificată

Funcţia 1Ch returnează informaţii despre dimensiunea şi tipul unităţii specificate.

Pentru apelul funcţiei, se folosesc regiştrii:

AH-încărcat cu 1Ch.
DL-încărcat cu numărul unităţii (0 = implicită, 1 = A, etc).

Funcţia returnează următoarele:

- în AL - numărul de sectoare pe unitate de alocare a discului;
- în CX - numărul de octeţi pe sector;
- în DX - numărul de unităţi de alocare ale discului pe unitate;
- în DS:BX - pointer la octetul de identificare FAT (acesta redă tipul unităţii).

Observaţii:

1. Pentru informaţii mai sigure despre tipul unităţii, se recomandă utilizarea INT 25h pentru citirea Blocului Parametrilor BIOS (BPB) din înregistrarea de iniţializare (sectorul de boot).
2. Pentru informaţii suplimentare, a se vedea funcţiile 1Bh, 32h, 36h şi FAT.

ATENŢIE!

Apelul acestei funcţii distruge registrul DS.

2.30. Funcţia 1Dh - Rezervată

2.31. Funcţia 1Eh - Rezervată

2.32. Funcţia 1Fh - Rezervată

2.33. Funcţia 20h - Rezervată

2.34. Funcţia 21h - Citirea în acces direct, utilizînd FCB

Funcţia 21h citeşte un articol dintr-un fişier în memorie (la adresa DTA curent), utilizînd un FCB al cărui deplasament faţă de adresa segment din registrul DS se află în registrul DX. Articolul de citit este specificat de cîmpurile Articol în acces direct (deplasament 21h) şi Dimensiune articol (deplasament 0Eh) din FCB. Cîmpurile Bloc curent şi Articol curent sînt poziţionate automat astfel încît să se potrivească cu cîmpul Articol în acces direct.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 21h;
DS:DX- încărcat cu un pointer la un FCB deschis.

Funcţia returnează în registrul AL următoarele:

00h- citire cu succes;
01h- s-a găsit EOF, fără a se citi date;
02h- DTA prea mic, se anulează citirea;
03h- s-a găsit EOF înaintea citirii unui articol întreg (se face completare cu zerouri).

Observaţii:

1. Cîmpul Articol în acces direct nu este actualizat.

2. În noile versiuni, este de preferat utilizarea funcţiei 3Fh.

3. Pentru informaţii suplimentare, a se vedea funcţiile 14h, 1Ah, 27h, 3Fh şi FCB.

2.35. Funcţia 22h - Scrierea în acces direct, utilizînd FCB

Funcţia 22h scrie un articol din memorie (de la adresa DTA curent) într-un fişier, utilizînd un FCB al cărui deplasamemt faţă de adresa segment din registrul DS se află în registrul DX. Articolul de scris este specificat de cîmpurile Articol în acces direct (deplasament 21h) şi Dimensiune articol (deplasament 0Eh) din FCB. Cîmpurile Bloc curent şi Articol curent sînt poziţionate automat astfel încît să se potrivească cu cîmpul Articol în acces direct.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 22h;
DS:DX- încărcat cu un pointer la un FCB deschis.

Funcţia returnează în registrul AL următoarele:

00h- citire cu succes;
01h- s-a găsit EOF, fără a se citi date;
02h- DTA prea mic, se anulează citirea.

Observaţii:

1. Cîmpul Articol în acces direct nu este actualizat.

2. Dacă cîmpul Dimensiune articol este mai mic decît un sector, datele sînt stocate într-o zonă tampon, fiind scris pe disc cînd se atinge dimensiunea unui sector, sau cînd se închide fişierul sau se face un apel al funcţiei 0Dh.

3. În fişierele "read-only" nu se va scrie.

4. În noile versiuni, este de preferat utilizarea funcţiei 40h.

5. Pentru informaţii suplimentare, a se vedea funcţiile 15h, 1Ah, 28h, 40h şi FCB.

2.36. Funcţia 23h - Obţinerea dimensiunii unui fişier, utilizînd FCB

Funcţia 23h returnează în cîmpul Articol în acces direct (deplasament 21h din FCB) dimensiunea unui fişier (în număr de articole, cu rotunjire superioarară), utilizînd un FCB al cărui deplasamemt faţă de adresa segment din registrul DS se află în registrul DX.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 23h;
DS:DX- încărcat cu un pointer la un FCB nedeschis.

Funcţia returnează în registrul AL valoarea 00h dacă s-a găsit fişierul specificat, respectiv FFh în caz contrar.

Observaţii:

1. Înainte de apel, trebuie poziţionat cîmpul Dimensiune articol (deplasament 0E din FCB). Dacă el este poziţionat la 1, se obţine dimensiunea în octeţi.

2. În noile versiuni, este de preferat utilizarea funcţiei 42h.

2.37. Funcţia 24h - Poziţionarea cîmpului Articol în acces direct, utilizînd FCB

Funcţia 24h poziţionează cîmpul Articol în acces direct (deplasament 21h) din FCB la adresa de fişier specificată de cîmpurile Bloc curent (deplasament 0Ch) şi Articol curent (deplasament 20h) din FCB, utilizînd un FCB al cărui deplasamemt faţă de adresa segment din registrul DS se află în registrul DX.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 24h;
DS:DX- încărcat cu un pointer la un FCB deschis.

Funcţia nu returnează nimic.

Observaţii:

1. Se recomandă utilizarea funcţiei înaintea unor citiri/scrieri în acces direct (funcţiile 21h, 22h, 27h, 28h).

2. În noile versiuni, este de preferat utilizarea funcţiei 42h.

2.38. Funcţia 25h - Poziţionarea vectorului de întrerupere

Funcţia 25h poziţionează adresa unei întreruperi în tabela vectorilor de întrerupere.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 25h;
AL - încărcat cu numărul întreruperii;
DS:DX- încărcat cu un pointer la rutina dorită de tratare a întreruperii.

Observaţie:

Se recomandă salvarea înainte de apel a vectorului respectiv de întrerupere (folosind funcţia 35h), astfel încît programul să poată reîncărca la ieşire adresa rutinei originale.

2.39. Funcţia 26h - Crearea unui PSP

Funcţia 26h creează un nou PSP (Program Segment Prefix). PSP-ul curent (256 octeţi) este copiat la adresa DX:0000, iar noul PSP este actualizat cu adresele rutinelor de tratare a întreruperilor INT 22h, INT 23h şi INT 24h. Este actualizat cîmpul cu informaţii despre memorie (locaţia 06h din noul PSP).

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 26h;
DX - încărcat cu adresa de segment a noului PSP.

Funcţia nu returnează nimic.

Observații:

1. În noile versiuni, este de preferat utilizarea funcției 4B00h.

2. Pentru informații suplimentare, a se vedea funcția 4Bh.

2.40. Funcția 27h - Citirea multiplă în acces direct, utilizînd FCB

Funcția 27h citește unul sau mai multe articole dintr-un fișier în memorie (la adresa DTA curent), utilizînd un FCB al cărui deplasament față de adresa segment din registrul DS se află în registrul DX. Citirea începe de la articolul specificat în cîmpul Articol în acces direct (deplasament 21h) din FCB. Numărul de octeți de citit este calculat înmulțind cîmpul Dimensiune articol (deplasament 0Eh) din FCB cu valoarea specificată în registrul CX.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 27h;
DS:DX- încărcat cu un pointer la un FCB deschis;
CX - numărul de articole de citit (dacă este egal cu zero, nu se citește nici un articol).

Funcția returnează următoarele:

- În registrul AL:
  - 00h - citire cu succes;
  - 01h - s-a găsit EOF, fără a se citi date;
  - 02h - DTA prea mic, se anulează citirea;
  - 03h - s-a găsit EOF înaintea citirii unui articol întreg (se face completare cu zerouri).

- În registrul CX: numărul de articole citite.

Observații:

1. Spre deosebire de funcția 21h, această funcție poate citi articole multiple, iar după citire, cîmpurile Articol în acces direct, Bloc curent și Articol curent din FCB sînt actualizate pentru a indica articolul următor.

2. Înainte de apel, se recomandă utilizarea funcţiei 24h.

3. În noile versiuni, este de preferat utilizarea uneia din funcţiile 42h sau 3Fh.

4. Pentru informaţii suplimentare, a se vedea funcţiile 14h, 1Ah, 21h, 24h, 3Fh, 42h şi FCB.

2.41. Funcţia 28h - Scrierea multiplă în acces direct, utilizînd FCB

Funcţia 28h scrie unul sau mai multe articole din memorie (de la adresa DTA curent) într-un fişier, utilizînd un FCB al cărui deplasament faţă de adresa segment din registrul DS se află în registrul DX. Scrierea începe de la articolul specificat în cîmpul Articol în acces direct (deplasament 21h) din FCB. Numărul de octeţi de citit este calculat înmulţind cîmpul Dimensiune articol (deplasament 0Eh) din FCB cu valoarea specificată în registrul CX. Dacă nu se scrie nimic (CX=0), DOS poziţionează cîmpul Dimensiune fişier (1Ch) din intrarea în director la valoare din cîmpul Articol în acces direct (deplasament 21h) din FCB, alocînd sau eliberînd unităţi de alocare ale discului.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 28h;
DS:DX - încărcat cu un pointer la un FCB deschis;
CX - numărul de articole de scris (dacă este egal cu zero, nu se scrie nici un articol).

Funcţia returnează următoarele:

- În registrul AL:
  - 00h - citire cu succes;
  - 01h - s-a găsit EOF, fără a se citi date;
  - 02h - DTA prea mic, se anulează citirea;
  - 03h - s-a găsit EOF înaintea citirii unui articol întreg (se face completare cu zerouri).

- În registrul CX: numărul de articole scrise.

Observaţii:

1. Spre deosebire de funcţia 21h, această funcţie poate scrie articole multiple, iar după scriere, Cîmpurile Articol în acces direct, Bloc curent şi Articol curent din FCB sînt actualizate pentru a indica articolul următor.

2. Dacă cîmpul Dimensiune articol este mai mic decît un sector, datele sînt stocate într-o zonă tampon, fiind scris pe disc cînd se atinge dimensiunea unui sector, sau cînd se închide fişierul sau se face un apel al funcţiei 0Dh.

3. În fişierele "read-only" nu se va scrie.

4. În noile versiuni, este de preferat utilizarea funcţiei 42h sau 40h.

5. Pentru informaţii suplimentare, a se vedea funcţiile 15h, 1Ah, 22h, 24h, 40h şi FCB.

2.42. Funcţia 29h - Analiza unui nume de fişier

Funcţia 29h analizează un şir al cărui deplasament faţă de adresa segment din DS se află în SI. Dacă se găseşte un specificator valid de fişier (dispozitiv: nume_fişier.extensie), va fi creat un FCB nedeschis într-o zonă de memorie al cărei deplasament faţă de adresa segment din ES se află în DI. Funcţia permite unui program să selecteze un dispozitiv, nume de fişier sau extensie implicită, sau să le înlocuiască cu cele dorite de utilizator.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 29h;
AL - încărcat cu biţii de control pentru analiză, cu următoarele semnificaţii:

Bit	Valoare	Semnificaţie
0	0	Se opreşte analiza dacă se întîlneşte un separator de fişier.

1	Se ignoră separatorii.
0	Poziţionează la 0 numărul dispozitivului, dacă nu este specificat în şir.
1	Lasă neschimbat numărul dispozitivului, dacă nu este specificat în şir.
0	Poziţionează numele fişierului la 8 blancuri, dacă nu este specificat în şir.
1	Lasă neschimbat numele fişierului, dacă nu este specificat în şir.
0	Poziţionează extensia la 3 blancuri, dacă nu este specificată în şir.
1	Lasă neschimbată extensia, dacă nu este specificată în şir.

DS:SI - pointer la şirul de analizat;
ES:DI- pointer la zona tampon pentru un FCB nedeschis.

Funcţia returnează următoarele:

- În registrul AL:

FFh - eşec;
00h - succes, nu s-au întilnit caracterele '*' sau '?';
01h - succes, s-au întilnit caracterele '*' sau '?'.

- În registrul DS:SI - pointer la octetul imediat următor irului de analizat.

- În registrul ES:DI:

în caz de succes - pointer la un FCB nedeschis;
în caz de eşec - al doilea octet de la adresa pointată (ES:DI + 1) conţine un contor blanc.



Observații:

1. Caracterul '*' în nume_fişier sau extensie este înlocuit automat cu un număr corespunzător de caractere '?'. 1

2. Separatori de nume de fişier sînt:

   : . ; , = + / " [ ] \ < > | blanc tab 2

   Terminatorii de nume de fişier includ orice separator, plus un caracter de control. Ei nu pot apărea în numele de fişier.

3. Funcția nu poate trata nume de căi. 3

2.43. Funcția 2Ah - Obținerea datei sistem curente

Funcția 2Ah returnează data sistem curentă, inclusiv ziua din săptămînă.

Pentru apelul funcției, se foloseşte registrul:

AH - încărcat cu 2Ah.

Funcția returnează următoarele:

- în registrul AL -	ziua din săptămînă (0=duminică, ..., 6=sîmbătă);
- în registrul CX -	anul curent (1980 - 2098);
- în registrul DH -	luna curentă (1 - 12);
- în registrul DL -	ziua curentă (1 -31).

Observații:

1. Manualele DOS 2.00 şi 2.10 specifică returnarea zilei din săptămînă în registrul AL de către funcția 2Ch. Acest lucru este inexact, el fiind realizat de funcția de față.

2. Pentru informații suplimentare, a se vedea funcțiile 2Bh, 2Ch, 2Dh.

2.44. Funcţia 2Bh - Poziţionarea datei sistem curente

Funcţia 2Bh poziţionează data sistem curentă.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 2Bh;
CX - încărcat cu anul curent (1980 - 2098);
DH - încărcat cu luna curentă (1 - 12);
DL - încărcat cu ziua curentă (1 - 31).

Funcţia returnează în registrul AH valoarea 00h dacă s-a specificat o dată validă, respectiv FFh în caz contrar (nu are loc poziţionarea).

Pentru informaţii suplimentare, a se vedea funcţiile 2Ah, 2Ch, 2Dh.

2.45. Funcţia 2Ch - Obţinerea orei sistem curente

Funcţia 2Ch returnează ora sistem curentă.

Pentru apelul funcţiei, se foloseşte registrul:

AH - încărcat cu 2Ch.

Funcţia returnează următoarele:

- în registrul CH - ora curentă (0 - 23);
- în registrul CL - minutul curent (0 - 59);
- în registrul DH - secunda curentă (0 - 59);
- în registrul DL - sutimea de secundă curentă (0 - 99).

Observaţii:

1. În majoritatea sistemelor, datorită frecvenţei de tact de 18.2 MHz, valoarea din DL va avea o precizie de 1/18.2 secunde, în loc de 1/100.

2. În sistemele în care ceasul nu urmăreşte sutimile de secundă, valoarea din DL este necunoscută (probabil 0).

3. Pentru informaţii suplimentare, a se vedea funcţiile 2Ah, 2Bh, 2Dh şi INT 08h, INT 15h.

2.46. Funcţia 2Dh - Poziţionarea orei sistem curente

Funcţia 2Dh poziţionează ora sistem curentă.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 2Dh;
CH - încărcat cu ora curentă (0 - 23);
CL - încărcat cu minutul curent (0 - 59);
DH - încărcat cu secunda curentă (0 - 59);
DL - încărcat cu sutimea de secundă curentă (1 - 100).

Funcţia returnează în registrul AL valoarea 00h dacă s-a specificat o oră validă, respectiv FFh în caz contrar (nu are loc poziţionarea).

Observaţii:

1. În sistemele în care ceasul nu urmăreşte sutimile de secundă, valoarea din DL este irelevantă.

2. Pentru informaţii suplimentare, a se vedea funcţiile 2Ah, 2Bh, 2Ch şi INT 08h, INT 15h.

2.47. Funcţia 2Eh- Poziţionarea/Repoziţionarea indicatorului VERIFY

Funcţia 2Eh specifică dacă DOS trebuie să verifice (prin citire înapoi) fiecare sector, după scrierea pe disc.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 2Eh;

AL - încărcat cu 00h pentru poziţionarea indicatorului pe 0 (VERIFY OFF - nu verifică);
încărcat cu 01h pentru poziţionarea indicatorului pe 1 (VERIFY ON - verifică).

Funcţia nu returnează nimic.

Observaţii:

1. Pentru AL = 01h, funcţia asigură o maximă integritate a datelor, dar încetineşte operaţia de scriere.

2. Starea indicatorului VERIFY se poate obţine cu funcţia 54h.

3. În versiunile DOS 1 şi 2, înainte de apel trebuie încărcat 00h în registrul DL.

4. Funcţia nu funcţionează pe discuri din reţea.

2.48. Funcţia 2Fh - Obţinerea adresei DTA curente

Funcţia 2Fh returnează adresa de segment de început pentru DTA curentă în ES şi deplasamentul în BX.

Pentru apelul funcţiei, se foloseşte registrul:

AH - încărcat cu 2Fh.

Funcţia returnează în ES:BX adresa DTA curentă.

Observaţii:

1. Se recomandă utilizarea funcţiei în proceduri recursive, pentru salvarea DTA.

2. Dacă adresa DTA nu este poziţionată explicit cu funcţia 1Ah, la deplasamentul 80h din PSP va fi pusă implicit de DOS o DTA de 128 octeţi.

ATENȚIE!

Funcția modifică registrul de segment ES.

2.49. Funcția 30h - Obținerea numărului versiunii DOS

Funcția returnează valoarea versiunii DOS active.

Pentru apelul funcției, se folosește registrul:

AH - încărcat cu 30h.

Funcția returnează următoarele:

- în registrul AL - numărul versiunii de bază;
- în registrul AH - numărul versiunii secundare.

Exemplu:

DOS 2.11 returnează 02h în AL și 0Bh în AH.

Observație:

Dacă se returnează 0 în AL, este vorba de o versiune DOS mai mică decît 2.0.

2.50. Funcția 31h - Păstrarea procesului

Funcția 31h realizează ieșirea în procesul părinte, astfel încît programul din care se iese sa rămînă rezident după terminare.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 31h;
AL - încărcat cu codul de retur (codul de eroare);
DX - încărcat cu dimensiunea memoriei rezidente, în paragrafe (un paragraf = 16 octeți).

Funcția nu returnează nimic.

Observații:

1. Funcția 31h înlocuiește INT 27h, avînd două îmbunătățiri: permite sa rămînă rezidenți mai mult de 64 Kocteți de memorie, și permite procesului care se termină să transmită un cod de retur.

2. Valoarea din DX trebuie să fie dimensiunea totala a memoriei care sa rămînă rezidentă, și nu doar dimensiunea segmentului de cod (a nu se uita cei 100h octeți de la începutul programelor .EXE).

3. Memoria specificată în DX este alocată, dar nu se eliberează memoria alocată prin funcția 48h de procesul care se termină.

4. Codul de retur poate fi obținut de procesul părinte prin funcția 4Dh și poate fi testat prin comanda IF cu ERRORLEVEL.

5. Pentru informații suplimentare, a se vedea funcțiile 00h, 4Dh și INT 20h, INT 27h.

ATENȚIE!

Această funcție nu închide fișierele deschise.

2.51. Funcția 32h - Rezervată

2.52. Funcția 33h - Poziționarea/Obținerea stării indicatorului Ctrl/C

Funcția 33h returnează sau poziționează starea verificării introducerii caracterului Ctrl/C (sau Ctrl/Break pentru sisteme compatibile IBM). Cînd starea este OFF (implicit), este verificată introducerea caracterului Ctrl/C doar în timpul operațiilor de intrare/ieșire cu echipamente standard (funcțiile 01h - 0Ch, exceptînd 06h și 07h). În versiunile DOS ulterioare lui 2.00, verificarea se poate extinde și la restul funcțiilor (exceptînd 06h și 07h), poziționînd această stare pe ON.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 33h;
AL - încărcat cu:
    00h - pentru obţinerea stării;
    01h - pentru poziţionarea stării.
DL - (dacă AL = 1) încărcat cu:
    0 = OFF;
    1 = ON.

Funcţia returnează următoarele:

- În registrul DL:

    0 = OFF;
    1 = ON.

- În registrul AL:

    FFh în caz de eroare (AL a fost diferit de 0 sau 1). În acest caz, starea verificării introducerii caracterului Ctrl/C nu este afectată.

Observaţii:

1. Starea verificării poate fi poziţionată cu comanda DOS BREAK, astfel încît se recomandă salvarea ei şi repoziţionarea cînd se revine în DOS.

2. Pentru informaţii suplimentare, a se vedea INT 23h.

2.53. Funcţia 34h - Rezervată

2.54. Funcţia 35h - Obţinerea vectorului de întrerupere

Funcţia 35h returnează, din tabela vectorilor de întrerupere, adresa vectorului specificat, punînd în ES adresa de segment a rutinei de tratare a întreruperii, iar în BX deplasamentul.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 35h;
AL - încărcat cu numărul întreruperii.

Funcţia returnează în ES:BX un pointer la rutina de tratare a întreruperii.

Observaţie:

Pentru a evita problemele de compatibilitate, se recomandă ca un vector de întrerupere să nu fie citit direct din memorie sau poziţionat prin scriere direct în memorie. Se va utiliza funcţia 35h pentru a obţine vectorul de întrerupere şi funcţia 25h pentru a-l poziţiona.

2.55. Funcţia 36h - Obţinerea spaţiului liber pe disc

Funcţia 36h returnează numărul unităţilor de alocare ale discului disponibile pe unitate şi alte informaţii care permit calcularea numărului de octeţi disponibili.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 36h;
DL - încărcat cu numărul unităţii (0 = implicit, 1 = A, etc).

Funcţia returnează următoarele:

- În registrul AX - numărul de sectoare pe unitate de alocare a discului, respectiv FFFFh dacă s-a specificat un număr invalid de unitate;
- În registrul BX - numărul unităţilor de alocare ale discului disponibile;
- În registrul CX - numărul de octeţi pe sector;
- În registrul DX - numărul de unităţi de alocare ale discului pe unitate.

Observații:

1. Funcția 36h înlocuiește funcțiile 1Bh și 1Ch din versiunile DOS vechi.

2. Spațiul liber în octeți =(AX * BX * CX).
   Spațiul total în octeți =(AX * CX * DX).

3. Pentru informații suplimentare, a se vedea funcțiile 1Bh, 1Ch, 44h.

2.56. Funcția 37h - Rezervată

2.57. Funcția 38h - Obținerea/Poziționarea informațiilor dependente de țară

Funcția 38h returnează informațiile dependente de țară (într-o zonă de memorie cu adresa de segment în DS și deplasamentul în DX) pe care DOS le folosește pentru controlul tastaturii și afișajului, sau poziționează țara curentă.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 38h.
AL - încărcat cu:
- 00h - țară curentă;
- 01h - FEh- codul țării (<=255);
- FFh - codul țării se află în BX (>255).

BX - (dacă AH = FFh) - codul țării (>255).

I. Pentru obținerea informațiilor:

DS:DX - încărcat cu un pointer la o zonă de memorie de 34 octeți.

II. Pentru poziționarea codului țării:

DX - încărcat cu FFFFh.

Funcţia returnează următoarele:

- Dacă este poziţionat indicatorul CF:
    - în registrul AH - eroare 02h - cod ţară invalid.

- Dacă nu este poziţionat indicatorul CF:
    - în registrul BX - codul ţării;
    - în registrul DS:DX (cazul I.) - informaţiile dependente de ţară.

Informaţiile returnate în zona pointată de DS:DX au următoarea structură:

Deplasament	Conţinut	Lungime în octeţi
00h	Format dată	2 (cuvînt)
02h	Simbol etalon monetar	5 (şir ASCIIZ)
07h	Separator de mii	2 (şir ASCIIZ)
09h	Separator zecimal	2 (şir ASCIIZ)
0Bh	Separator de dată	2 (şir ASCIIZ)
0Dh	Separator de oră	2 (şir ASCIIZ)
0Fh	Format etalon monetar	1
10h	Număr de cifre zecimale semnificative în etalonul monetar	1
11h	Format oră	1
12h	Pointer la o procedură de tip FAR	4 (dublu cuvînt)
16h	Separator în lista de date	2 (şir ASCII)
18h	Rezervaţi	10

Format dată:

0 = SUA (ll/zz/aa);
1 = Europa (zz/ll/aa)
2 = Japonia (aa/ll/zz).

Format etalon monetar - este un cîmp de biţi:

Valoare	Semnificaţie
00h	Simbolul precede valoarea, fără spaţiu.
01h	Simbolul urmează valoarea, fără spaţiu.
02h	Simbolul precede valoarea, un spaţiu între simbol şi valoare.
03h	Simbolul urmează valoarea, un spaţiu între simbol şi valoare.
04h	Simbolul înlocuieşte separatorul zecimal (doar în versiunile DOS >=3.2).

Format oră:

0 = ceas cu 12 ore (am/pm);
1 = ceas cu 24 ore.

Procedura FAR (deplasament 12h) realizează prelucrări specifice ţării, transformînd din mic în mare un caracter cu codul între 80h şi FFh. Apelul ei se face cu caracterul în AL. Dacă există codul respectiv pentru majusculă, el este returnat în AL. Dacă nu există sau caracterul este mai mic decît 80h, AL nu se modifică. Sînt afectaţi registrul AL şi indicatoarele.

Observaţii

1. Codul ţării este de obicei prefixul telefonic internaţional.

2. Versiunile DOS 2.x permit obţinerea informaţiilor doar pentru ţara curentă, şi nu permit poziţionarea codului ţării.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 38h;
AL - încărcat cu 00h;
DS:DX - încărcat cu un pointer la o zonă de memorie de 32 octeţi.

Funcţia returnează următoarele:



- Dacă este poziţionat indicatorul CF:
  - în registrul AH - eroare 02h : cod ţară invalid.

- Dacă nu este poziţionat indicatorul CF:
  - în registrul DS:DX - informaţiile dependente de ţară.

Informaţiile returnate în DS:DX au următoarea structură:

Deplasament	Conţinut	Lungime în octeţi
00h	Format dată/oră 2	(cuvînt)
02h	Simbol monetar curent	2 (şir ASCIIZ)
04h	Separator de mii	2 (şir ASCIIZ)
06h	Separator zecimal	2 (şir ASCIIZ)
08h	Rezervaţi	24

Format dată/oră:

0 = SUA (oo:mm:ss ll/zz/aa);
1 = Europa (oo:mm:ss zz/ll/aa);
2 = Japonia (oo:mm:ss aa/ll/zz).

2.58. Funcţia 39h - Crearea unui nou director (MKDIR)

Funcţia 39h creează un nou subdirector, utilizînd ca şi nume de cale un şir ASCIIZ (maxim 64 caractere) cu deplasamentul faţă de adresa registru din DS aflat în DX.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 39h;
DS:DX - încărcat cu un pointer la numele noului director ("d:\nume_cale\nume_director").

Dacă după retur indicatorul CF este poziţionat, directorul nu a fost creeat şi funcţia returnează în AX un cod de eroare:

3 - nu s-a găsit calea specificată;

5 - acces interzis (nu există spaţiu, există deja un director cu acelaşi nume sau s-a specificat un nume de unitate).

Observaţii:

1. Dacă se omite numele dispozitivului şi/sau al căii, se vor considera cele implicite.

2. Pentru informaţii suplimentare, a se vedea funcţiile 3Ah, 3Bh, 47h, 59h.

2.59. Funcţia 3Ah - Ştergerea unui director (RMDIR)

Funcţia 3Ah şterge un subdirector, utilizînd ca şi nume de cale un şir ASCIIZ (maxim 64 caractere) cu deplasamentul faţă de adresa registru din DS aflat în DX.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 3Ah;
DS:DX - încărcat cu un pointer la numele directorului ("d:\nume_cale\nume_director").

Dacă după retur indicatorul CF este poziţionat, directorul nu a fost şters şi funcţia returnează în AX un cod de eroare:

3 - nu s-a găsit calea specificată;
5 - acces interzis (directorul nu este gol, calea nu specifică un director valid sau specifică directorul rădăcină);
6 - calea specifică directorul curent.

Observaţii:

1. Dacă se omite numele dispozitivului şi/sau al căii, se vor considera cele implicite.

2. Directorul trebuie să fie gol (să nu conţină fişiere sau alte subdirectoare).

3. Directorul rădăcină şi cel curent nu pot fi şterse.

4. Pentru informaţii suplimentare, a se vedea funcţiile 39h, 3Bh, 47h, 59h.

2.60. Funcţia 3Bh - Schimbarea directorului (CHDIR)

Funcţia 3Bh schimbă directorul curent la cel specificat, utilizînd ca şi nume de cale un şir ASCIIZ (maxim 64 caractere) cu deplasamentul faţă de adresa registru din DS aflat în DX.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 3Bh;
DS:DX - încărcat cu un pointer la numele noului director ("d:\nume_cale\nume_director").

Dacă după retur indicatorul CF este poziţionat, directorul nu a fost schimbat şi funcţia returnează în AX codul de eroare 3 (nu s-a găsit calea).

Observaţii:

1. Dacă se omite numele dispozitivului şi/sau al căii, se vor considera cele implicite.

2. Pentru informaţii suplimentare, a se vedea funcţiile 39h, 3Ah, 47h, 59h.

2.61. Funcţia 3Ch - Crearea unui fişier (CREAT)

Funcţia 3Ch creează un fişier cu numele dat de şirul ASCIIZ ("d:\nume_cale\specif_fişier") avînd deplasamentul faţă de adresa segment din DS în DX, deschizîndu-l în mod scriere/citire.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 3Ch;



DS:DX - încărcat cu un pointer la numele fişierului (şir ASCIIZ);
CX - încărcat cu atributul fişierului:
00h - normal;
01h - "read-only";
02h - ascuns;
04h - sistem.

Dacă după retur nu este poziţionat indicatorul CF, funcţia returnează în AX identificatorul logic al fişierului, respectiv un cod de eroare în caz contrar:

3 - nu s-a găsit calea;
4 - prea multe fişiere deschise;
5 - acces interzis (director plin sau există un fişier cu acelaşi nume, dar cu atribute mai restrictive).

Observaţii:

1. Dacă fişierul specificat nu există, el va fi creat şi i se va asigna primul identificator logic liber. Dacă există, el este trunchiat la lungime 0 (şters).

2. Încercarea de a deschide un fişier "read-only" va genera o eroare. Totuşi, se pot scrie informaţii într-un fişier creat în mod "read-only".

3. Dacă nu se specifică unitatea şi/sau calea, se vor lua cele implicite.

4. Pentru informaţii suplimentare, a se vedea funcţiile 16h, 43h, 5Ah, 5Bh, 59h.

2.62. Funcţia 3Dh - Deschiderea unui fişier, utilizînd identificator logic

Funcţia 3Dh deschide orice fişier existent, inclusiv cele ascunse sau sistem, cu numele dat de şirul ASCIIZ ("d:\nume_cale\specif_fişier") avînd deplasamentul faţă de adresa

segment din DS în DX. Pointerul de scriere/citire este poziționat la primul octet al fișierului.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 3Dh;
AL - încărcat cu codul de acces (a se vedea mai jos);
DS:DX - încărcat cu un pointer la numele fișierului (șir ASCIIZ).

Dacă după retur nu este poziționat indicatorul CF, funcția returnează în AX identificatorul logic al fișierului, respectiv un cod de eroare în caz contrar:

1 - număr funcție invalid;
2 - nu s-a găsit fișierul;
3 - nu s-a găsit calea;
4 - prea multe fișiere deschise;
5 - acces interzis;
12 - mod de deschidere invalid.

Codul de acces din AL:

Biți	Mod
7 6 5 4 3 2 1 0	
. . . . . A A A	Mod acces ("read/write" în DOS 2.x)
. . . . R . . .	Rezervat (0 în DOS 2.x)
. P P P	Mod partajare (0 în DOS 2.x)
M	Indicator de moștenire (0 în DOS 2.x)

Bitul de moștenire specifică dacă fișierul este moștenit de către un proces fiu creat cu funcția 4Bh (0), sau nu (1). Dacă da, procesul fiu are acces la fișier în mod automat.

Modul partajare specifică ce accese au alte procese la fișierul deschis:

Biți 4 - 6	Mod partajare	Descriere
000	Compatibilitate	Orice proces poate deschide fișierul de oricîte ori. Eșuează dacă a fost deschis în oricare alt mod partajare.
001	Interzis orice	Eșuează dacă fișierul a fost deschis în mod compatibilitate sau în acces "read/write", chiar de către procesul curent.
010	Interzisă scrierea	Eșuează dacă fișierul a fost deschis în mod compatibilitate sau în acces scriere de orice alt proces.
011	Interzisă citirea	Eșuează dacă fișierul a fost deschis în mod compatibilitate sau în acces citire de orice alt proces.
100	Interzis nimic	Eșuează dacă fișierul a fost deschis în mod compatibilitate de orice alt proces.

Modul acces specifică cum va fi utilizat fișierul:

Biți 0 - 2	Permisă	Descriere
000	Citirea	Eșuează dacă fișierul a fost deschis în mod "Interzisă citirea" sau "Interzis orice".
001	Scrierea	Eșuează dacă fișierul a fost deschis în mod "Interzisă scrierea" sau "Interzis orice".
010	Scrierea/Citirea	Eșúează dacă fișierul a fost deschis în mod "Interzisă scrierea", "Interzisă citirea" sau "Interzis orice".

Observații:

1. Un fișier este considerat a fi în mod "compatibilitate" dacă este deschis:

a. printr-o funcţie care lucrează cu FCB;
b. printr-o funcţie CREATE;
c. printr-o funcţie care lucrează cu identificator logic de fişier, dacă se specifică acest mod.

2. Dacă nu se specifică unitatea şi/sau calea, se vor lua cele implicite.

3. Modul partajare este utilizabil doar după execuţia comenzii DOS SHARE, care încarcă suportul de partajare a fişierelor.

4. Pentru informaţii suplimentare, a se vedea funcţiile 0Fh, 16h, 3Ch, 5Ah, 5Bh, 59h.

2.63. Funcţia 3Eh - Închiderea unui fişier, utilizînd identificator logic

Funcţia 3Eh goleşte toate zonele tampon ale fişierului, închide fişierul (deschis cu una din funcţiile 3Dh sau 3Ch), eliberează identificatorul logic şi actualizează directorul.

Pentrul apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 3Eh;
BX - încărcat cu identificatorul logic al fişierului.

Dacă este poziţionat indicatorul CF, funcţia returnează în AX codul de eroare 6 - identificator logic invalid.

Observaţii:

1. În cazul unei depăşiri a numărului de fişiere pe care un program le poate deschide simultan (numărul implicit de identificatoare logice este 8, iar maxim 20), se poate închide unul implicit (de exemplu, identificatorul logic 3, Standard AUX).

2. Pentru informaţii suplimentare, a se vedea funcţiile 10h, 3Dh, 59h.

2.64. Funcția 3Fh - Citirea dintr-un fișier sau de la un dispozitiv, utilizînd identificator logic

Funcția 3Fh citește dintr-un fișier sau de la un dispozitiv asociat identificatorul logic specificat, într-o zonă tampon al cărei deplasament față de adresa segment din DS se află în DX. Datele se citesc de la poziția curentă a pointerului de scriere/citire, după care acesta este incrementat cu numărul de octeți citiți. Citind de la intrarea standard, aceasta poate fi redirectată.

Pentru apelul funcției, se folosesc regiștrii:

- AH - încărcat cu 3Fh;
- BX - încărcat cu identificatorul logic;
- CX - încărcat cu numărul de octeți de citit;
- DS:DX- încărcat cu un pointer la zona tampon de scriere.

Dacă după retur nu este poziționat indicatorul CF, funcția returnează în AX numărul de octeți cititi, respectiv un cod de eroare în caz contrar:

- 5 - acces interzis;
- 6 - identificator logic nedeschis sau invalid.

Observații:

1. Funcția 3Fh înlocuiește funcțiile complicate care lucrează cu FCB, combinînd accesul secvențial cu cel direct și permițînd o structurare a datelor proprie utilizatorului.

2. Dacă CF nu este poziționat și AX este diferit de CX, există două posibilități:

   a. AX = 0 - s-a încercat citirea începînd cu EOF;
   b. 0 < AX < CX - s-a ajuns la EOF înainte de a se citi numărul de octeți din CX.

3. Dacă se citește de la un dispozitiv, se va citi sau numărul de octeți din CX, sau pînă la primul <CR>.

4. Pentru informații suplimentare, a se vedea funcțiile 14h, 21h, 27h, 42h, 59h.

2.65. Funcția 40h - Scrierea într-un fișier sau la un dispozitiv, utilizînd identificator logic

Funcția 40h scrie într-un fișier sau la un dispozitiv asociat identificatorului logic specificat datele dintr-o zonă tampon al cărei deplasament față de adresa segment din DS se află în DX. Datele sînt scrise de la poziția curentă a pointerului de scriere/citire, după care acesta este incrementat cu numărul de octeți scriși. Scriind la ieșirea standard, aceasta poate fi redirectată.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 40h;
DX - încărcat cu identificatorul logic;
CX - încărcat cu numărul de octeți de scris;
DS:DX- încărcat cu un pointer la zona tampon de citire.

Dacă după retur nu este poziționat indicatorul CF, funcția returnează în AX numărul de octeți scriși, respectiv un cod de eroare în caz contrar:

5 - acces interzis ("read-only");
6 - identificator logic nedeschis sau invalid.

Observații:

1. Funcția 40h înlocuiește funcțiile complicate care lucrează cu FCB, combinînd accesul secvențial cu cel direct și permițînd o structurare a datelor proprie utilizatorului.

2. Apelînd funcția cu CX = 0, fișierul se poate trunchia, noul EOF fiind pe poziția pointerului de scriere/citire.

3. Se recomandă întotdeauna verificarea valorii din AX după scriere. Dacă AX < CX, a apărut o eroare, chiar dacă CF nu este poziționat (de obicei disc plin).

4. Pentru informații suplimentare, a se vedea funcțiile: 15h, 22h, 28h, 42h 59h.

2.66. Funcția 41h - Ștergerea unui fișier (UNLINK)

Funcția 41h șterge intrarea din director pentru fișierul al cărui nume dat ca șir ASCIIZ ("d:\nume_cale\specif_fișier") are deplasamentul față de adresa segment din DS în DX. În șir nu pot apărea caracterele '?' și '*'.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 41h;
DS:DX- încărcat cu un pointer la numele fișierului (șir ASCIIZ).

Dacă după retur indicatorul CF este poziționat, funcția returnează în AX un cod de eroare:

2 - nu s-a găsit fișierul;
3 - cale invalidă;
5 - acces interzis (s-a specificat un director sau un fișier "read-only").

Observații:

1. Dacă se omite unitatea și/sau calea, se vor lua cele implicite.

2. Pentru informații suplimentare, a se vedea funcțiile 13h, 43h, 59h.

2.67. Funcția 42h - Poziționarea pointerului de scriere/citire (LSEEK)

Funcția 42h schimbă poziția pointerului de scriere/citire a fișierului asociat cu identificatorul logic specificat.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 42h;
AL - încărcat cu codul metodei de poziționare:

0 - față de începutul fișierului;
1 - față de poziția curentă;
2 - față de sfîrșitul fișierului.

BX - încărcat cu identificatorul logic al fișierului;
CX:DX - încărcat cu distanța în octeți față de poziția dată în AL (deplasamentul este specificat ca un întreg cu semn pe 32 de biți). CX conține cuvîntul cel mai semnificativ.

Dacă după retur nu este poziționat indicatorul CF, funcția returnează în DX:AX noua locație a pointerului (întreg cu semn pe 32 de biți), respectiv un cod de eroare în AX în caz contrar:

1 - funcție invalidă (AL nu este 0, 1 sau 2);
6 - identificator logic invalid sau nedeschis.

Observații:

1. Pointerul se poate muta la o locație înainte de începutul fișierului sau după EOF. Nu apare eroare, dar în primul caz, următoarea operație de scriere/citire va eșua.

2. Fișierul poate fi extins mutînd pointerul după EOF și scriind unul sau mai mulți octeți de date.

3. Fișierul poate fi trunchiat mutînd pointerul la noul EOF și scriind zero octeți cu funcția 40h.

4. Se poate determina lungimea fișierului apelînd funcția cu 0 în CX:DX și 2 în AL.

5. Pentru informații suplimentare, a se vedea funcțiile 40h, 59h.

2.68. Funcţia 43h - Obţinerea/Poziţionarea atributelor fişierului (CHMOD)

Funcţia 42h returnează sau poziţionează atributele unui fişier specificat printr-un şir ASCIIZ ("d:\nume_cale\specif_fişier") al cărui deplasament faţă de adresa segment din DS se află în DX.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 43h;
AL - încărcat cu codul subfuncţiei:

00h - returnarea atributelor;
01h - poziţionarea atributelor.

CX - încărcat cu atributele de poziţionat (dacă AL = 1):

00h - normal;
01h - "read-only";
02h - ascuns;
04h - sistem;
20h - arhivă.

DS:DX- încărcat cu un pointer la numele fişierului (şir ASCIIZ).

Dacă după retur nu este poziţionat indicatorul CF, funcţia returnează în CX octetul de atribute (pentru cazul AL = 0), respectiv un cod de eroare în AX în caz contrar (indicatorul CF poziţionat):

1 - funcţie invalidă (AL nu este 0 sau 1);
2 - nu s-a găsit fişierul;
3 - nu s-a găsit calea;
5 - acces interzis (nu se poate schimba bitul de identificator de volum (08h) sau bitul de director (10h).

Observaţii:

1. Dacă se omite unitatea şi/sau calea, se vor lua cele implicite.

2. Pentru informaţii suplimentare, a se vedea funcţiile 3Ch, 59h.

2.69. Funcţia 44h - Control intrare/ieşire pentru dispozitive (IOCTL)

Funcţia 44h realizează un schimb de informaţii între o aplicaţie şi un dispozitiv sau un fişier.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 44h;
AL - încărcat cu codul subfuncţiei;
BX - încărcat cu identificatorul logic;
BL - încărcat cu numărul unităţii (0 = implicită, 1 = A, etc);
CX - încărcat cu numărul de octeţi de scris sau de citit;
DS:DX- încărcat cu un pointer la date sau la o zonă tampon.

Dacă după retur este poziţionat indicatorul CF, funcţia returnează un cod de eroare în AX, respectiv informaţii dependente de subfuncţia apelată în caz contrar.

Subfuncţia de apelat este determinată de valoarea din AL, după cum urmează:

AL	Subfuncţie
00h	Obţinerea informaţiilor despre dispozitiv.
01h	Poziţionarea informaţiilor despre dispozitiv.
02h	Citirea de octeţi de la canalul de control al dispozitivului în memorie (dispozitiv caracter).
03h	Scrierea de octeţi din memorie în canalul de control al dispozitivului (dispozitiv caracter).
04h	Citirea de octeţi de la canalul de control al dispozitivului în memorie (dispozitiv bloc).
05h	Scrierea de octeţi din memorie în canalul de control al dispozitivului (dispozitiv bloc).
06h	Obţinerea stării intrării.
07h	Obţinerea stării ieşirii.

08h	Se poate schimba dispozitivul bloc? (>= DOS 3.0).
09h	Dispozitivul logic este local sau la distanță? (>= DOS 3.1).
0Ah	Identificatorul logic este local sau la distanță? (>= DOS 3.1).
0Bh	Modificarea numărului de reîncercări (>= DOS 3.0).
0Ch	Cerere IOCTL generică pentru identificator logic(>= DOS 3.2).
0Dh	Cerere IOCTL generică pentru dispozitiv bloc (>= DOS 3.2).
0Eh	Obținerea dispozitivului logic (>= DOS 3.2).
0Fh	Selectarea dispozitivului logic (>= DOS 3.2).

Observații:

1. Doar subfuncțiile 00h, 06h, 07h sînt definite pentru fișiere.

2. Subfuncțiile 00h și 08h nu funcționează pe dispozitive rețea.

3. Subfuncția 0Bh necesită suportul de partajare a fișierelor, care se încarcă cu comanda DOS SHARE.

4. Pentru definirea noțiunilor de dispozitiv bloc și dispozitiv caracter, a se vedea subfuncția 00h.

5. Pentru informații suplimentare, a se vedea în continuare descrierea subfuncțiilor.

2.69.1. Funcția 4400h - IOCTL: Obținerea informațiilor despre dispozitiv

Subfuncția 00h returnează informații despre dispozitivul sau fișierul atașat identificatorului logic specificat.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 44h;
AL - încărcat cu 00h;
BX - încărcat cu identificatorul logic al dispozitivului sau fișierului.

Dacă este poziţionat indicatorul CF, funcţia returnează un cod de eroare în AX (6 - identificator logic invalid sau nedeschis), respectiv informaţiile dorite în DX, în caz contrar. Bitul 7 din DX este 1 dacă canalul este un dispozitiv caracter, şi 0 dacă canalul este un dispozitiv bloc (fişier disc).

DX pentru dispozitiv caracter (Bit 7 = 1):

Bit	Descriere
0	Poziţionat dacă este dispozitivul de intrare (consolă) standard.
1	Poziţionat dacă este dispozitivul de ieşire (consolă) standard.
2	Poziţionat dacă este dispozitivul nul.
3	Poziţionat dacă dispozitivul este un ceas.
4	Rezervat.
5	Poziţionat pentru mod binar, 0 pentru mod ASCII.
6	0 dacă EOF la intrare.
7	Poziţionat pentru a indica un dispozitiv (şi nu un fişier).
8 - 10	Rezervaţi.
11	Dispozitivul înţelege instrucţiuni "open/close".
12	Rezervat.
13	Dispozitivul suportă ieşirea pînă cînd devine ocupat.
14	Poziţionat dacă dispozitivul poate prelucra şiruri de control prin subfuncţiile 02h, 03h, 04h, 05h (acest bit poate fi doar citit, nu şi scris).
15	Rezervat.

DX pentru fişier disc (bit 7 = 0):

Bit	Descriere
0 - 5	Număr dispozitiv bloc (A = 0, B = 1, etc).
6	Poziţionat dacă fişierul a fost scris.
7	0 pentru a indica un dispozitiv bloc.
8 - 15	Rezervaţi.

Observaţii:

1. Biţii rezervaţi trebuie poziţionaţi la 0.



2. Dacă pentru dispozitive caracter bitul 5 este nul, vor fi verificate caracterele de control Ctrl/C, Ctrl/P, Ctrl/S şi Ctrl/Z. Pentru a fi citite ca şi date în loc de caractere de control, trebuie poziţionat bitul 5 şi apoi utilizată funcţia 33h sau comanda DOS BREAK pentru a invalida verificarea lui Ctrl/C.

3. Pentru informaţii suplimentare, a se vedea funcţiile 44h, 4401h.

2.69.2. Funcţia 4401h - IOCTL: Poziţionarea informaţiilor despre dispozitiv

Subfuncţia 01h poziţionează informaţiile despre dispozitivul ataşat identificatorului logic specificat.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 44h;
AL - încărcat cu 01h;
BX - încărcat cu identificatorul logic;
DH - încărcat cu 00h;
DL - încărcat cu informaţii despre dispozitiv.

Dacă după retur este poziţionat indicatorul CF, funcţia returnează un cod de eroare în AX:

1 - funcţie invalidă (DX nu este 00h);
6 - identificator logic invalid sau nedeschis.

În caz contrar, sînt poziţionate în registrul DX următoarele informaţii:

Bit	Descriere
0	Poziţionat dacă este dispozitivul de intrare (consolă) standard.
1	Poziţionat dacă este dispozitivul de ieşire (consolă) standard.
2	Poziţionat dacă este dispozitivul nul.
3	Poziţionat dacă dispozitivul este un ceas.

4 Rezervat.
5 Poziţionat pentru mod binar, 0 pentru mod ASCII.
6 0 dacă EOF la intrare.
7 Poziţionat pentru a indica un dispozitiv (şi nu un fişier).
8 - 15 00h.

Observaţii:

1. Funcţia 4401h nu funcţionează pe fişiere disc.

2. Pentru informaţii suplimentare, a se vedea funcţiile 44h, 4400h.

2.69.3. Funcţia 4402h - IOCTL: Citirea de la un dispozitiv caracter

Subfuncţia 02h citeşte octeţi de date de la canalul de control al dispozitivului caracter ataşat identificatorului logic specificat în zona de memorie pointată de DS:DX. Modul de citire al datelor depinde de driverul dispozitivului.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 44h;
AL - încărcat cu 02h;
BX - încărcat cu identificatorul logic al dispozitivului 0 = implicit, 1 = A, etc);
CX - încărcat cu numărul de octeţi de citit;
DS:DX- încărcat cu un pointer la zona tampon de scriere.

Dacă după retur nu este poziţionat indicatorul CF, funcţia returnează în AX numărul de octeţi transferaţi, respectiv un cod de eroare în caz contrar:

1 - dispozitivul nu permite şiruri de control;
6 - identificator logic invalid sau nedeschis.

Observaţii:

1. Driverul trebuie să permită interfeţe IOCTL (vezi funcţia 4400h, bit 14).

2. Pentru informaţii suplimentare, a se vedea funcţiile 44h, 40h.

2.69.4. Funcţia 4403h - IOCTL: Scrierea la un dispozitiv caracter

Subfuncţia 03h scrie octeţi de date din zona de memorie pointată de DS:DX în canalul de control al dispozitivului caracter ataşat identificatorului logic specificat. Modul de scriere al datelor depinde de driverul dispozitivului.

Pentru apelul funcţiei, se folosesc regiştrii:

AH	- încărcat cu 44h;
AL	- încărcat cu 03h;
BL	- încărcat cu identificatorul logic al dispozitivului (0 = A, 1 = B, etc.);
CX	- încărcat cu numărul de octeţi de scris;
DS:DX	- încărcat cu un pointer la zona de memorie cu date.

Dacă după retur nu este poziţionat indicatorul CF, funcţia returnează în AX numărul de octeţi transferaţi, respectiv un cod de eroare în caz contrar:

1 - dispozitivul nu permite şiruri de control;
6 - identificator logic invalid sau nedeschis.

Observaţii:

1. Driverul trebuie să permită interfeţe IOCTL (vezi funcţia 4400h, bit 14).

2. Pentru informaţii suplimentare, a se vedea funcţiile 44h, 4402h.

2.69.5. Funcţia 4404h - IOCTL: Citirea de la un dispozitiv bloc

Subfuncţia 04h citeşte octeţi de date de la canalul de control al dispozitivului bloc ataşat identificatorului logic specificat în zona

de memorie pointată de DX:DX. Modul de citire al datelor depinde de driverul dispozitivului.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 44h;
AL - încărcat cu 04h;
BL - încărcat cu identificatorul logic al dispozitivului (0 = implicit, 1 = A, etc.);
CX - încărcat cu numărul de octeţi de citit;
DS:DX- încărcat cu un pointer la zona tampon de scriere.

Dacă după retur nu este poziţionat indicatorul CF, funcţia returnează în AX numărul de octeţi transferaţi, respectiv un cod de eroare în caz contrar:

1 - dispozitivul nu permite şiruri de control;
5 - driver invalid în BL.

Observaţii:

1. Driverul trebuie să permită interfeţe IOCTL (vezi funcţia 4400h, bit 14).

2. Pentru informaţii suplimentare, a se vedea funcţiile 44h, 4405h.

2.69.6. Funcţia 4405h - IOCTL: Srierea la un dispozitiv bloc

Subfuncţia 05h scrie octeţi de date din zona de memorie pointată de DS:DX în canalul de control al dispozitivului bloc ataşat identificatorului logic specificat. Modul de scriere al datelor depinde de driverul dispozitivului.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 44h;
AL - încărcat cu 05h;
BL - încărcat cu identificatorul logic al dispozitivului (0 = implicit, 1 = A, etc.);

CX - încărcat cu numărul de octeţi de scris;
DS:DX- încărcat cu un pointer la zona tampon de citire.

Dacă după retur nu este poziţionat indicatorul CF, funcţia returnează în AX numărul de octeţi transferaţi, respectiv un cod de eroare în caz contrar:

1 - dispozitivul nu permite şiruri de control;
5 - driver invalid în BL.

Observaţii:

1. Driverul trebuie să permită interfeţe IOCTL (vezi funcţia 4400h, bit 14).

2. Pentru informaţii suplimentare, a se vedea funcţiile 44h, 4400h.

2.69.7. Funcţia 4406h - IOCTL: Obţinerea stării intrării

Subfuncţia 06h verifică dacă dispozitivul sau fişierul ataşat identificatorului logic specificat este pregătit pentru intrare.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 44h;
AL - încărcat cu 06h;
BX - încărcat cu identificatorul logic al dispozitivului sau fişierului.

Dacă după retur este poziţionat indicatorul CF, funcţia returnează un cod de eroare în AX:

5 - acces interzis;
6 - identificator logic invalid sau nedeschis.

În caz contrar, funcţia returnează în AL starea intrării:

00h - nu este pregătit (EOF la fişier, respectiv dispozitiv nepregătit).



FFh - pregătit (nu este EOF la fişier, respectiv dispozitiv pregătit).

Pentru informaţii suplimentare, a se vedea funcţiile 44h, 4407h.

2.69.8. Funcţia 4407h - IOCTL: Obţinerea stării ieşirii

Subfuncţia 07h verifică dacă dispozitivul sau fişierul ataşat identificatorului logic specificat este pregătit pentru ieşire.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 44h;
AL - încărcat cu 07h;
BX - încărcat cu identificatorul logic al dispozitivului sau fişierului.

Dacă după retur este poziţionat indicatorul CF, funcţia returnează în AX un cod de eroare:

5 - acces interzis;
6 - identificator logic invalid sau nedeschis.

În caz contrar, funcţia returnează în AL starea ieşirii:

00h - dispozitiv nepregătit;
- EOF la fişier (dar fiind fişier de ieşire, el este totuşi pregătit);
FFh - pregătit (nu este EOF la fişier, respectiv dispozitiv pregătit).

Pentru informaţii suplimentare, a se vedea funcţiile 44h, 4406h.

2.69.9. Funcţia 4408h - IOCTL: Dispozitivul se poate înlocui?

Subfuncţia 08h verifică dacă o unitate conţine un dispozitiv care poate fi schimbat (de exemplu, un disc flexibil) sau nu.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 44h;
AL - încărcat cu 08h;
BL - încărcat cu numărul unităţii: (0 = implicită, 1 = A, ...).

Dacă după retur este poziţionat indicatorul CF, funcţia returnează în AX un cod de eroare:

1 - funcţie invalidă;
2 - unitate invalidă.

În caz contrar, se returnează în AX una din valorile:

00h - poate fi schimbat;
01h - nu poate fi schimbat.

Observaţii:

1. Codul de eroare 1 (dispozitivul nu permite acest apel de funcţie) indică faptul că el nu poate fi schimbat.

2. Subfuncţia este disponibilă doar în versiunile DOS >= 3.0.

2.69.10. Funcţia 4409h- IOCTL: Dispozitivul logic se află la distanţă?

Subfuncţia 09h verifică dacă dispozitivul specificat este asociat unui director reţea (local) sau este redirectat unui server (la distanţă).

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 44h;
AL - încărcat cu 09h;
BL - încărcat cu numărul dispozitivului (0 = implicit, 1 = A, etc).

Dacă după retur este poziţionat indicatorul CF, funcţia returnează în AX un cod de eroare:

1 - funcţie invalidă;
15 - dispozitiv invalid.

În caz contrar, se returnează în DX următoarele:

- dispozitiv la distanţă - este poziţionat bitul 12, restul sînt 0.
- dispozitiv local - cuvîntul de atribuit din rutina de tratare a dispozitivului.

Observaţii:

1. Un program de aplicaţie nu ar trebui să testeze bitul 12, întrucît o aplicaţie nu trebuie să facă deosebire între fişiere (dispozitive) locale sau la distanţă.

2. Subfuncţia este disponibilă doar în versiunile DOS >= 3.1.

3. Pentru informaţii suplimentare, a se vedea funcţia 440Ah.

2.69.11. Funcţia 440Ah - IOCTL: Identificatorul logic se află la distanţă?

Subfuncţia 0Ah verifică dacă identificatorul logic specificat este asociat unui dispozitiv local sau la distanţă.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 44h;
AL - încărcat cu 0Ah;
BX - încărcat cu identificatorul logic.

Dacă după retur este poziţionat indicatorul CF, funcţia returnează în AX un cod de eroare:

1 - funcţie invalidă;
6 - identificator logic invalid.



În caz contrar, se returnează în DX următoarele:

- Identificator logic la distanță - este poziționat bitul 15, restul sînt 0.
- Identificator logic local - cuvîntul de atribut din headerul dispozitivului asociat.

Observații:

1. Un program de aplicație nu ar trebui să testeze bitul 15, întrucît o aplicație nu trebuie să facă deosebire între fișiere (dispozitive) locale sau la distanță.

2. Subfuncția este disponibilă doar în versiunile DOS >= 3.1.

3. Pentru informații suplimentare, a se vedea funcția 4409h.

2.69.12. Funcția 440Bh - IOCTL: Modificarea numărului de reîncercări

Subfuncția 0Bh specifică de cîte ori DOS trebuie să reîncerce o operație cu discul care a eșuat datorită unei violări a partajării fișierelor. Valoarea implicită este 3. După ce se efectuează numărul de reîncercări specificat, DOS generează INT 24h pentru procesul apelant.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 44h;
AL - încărcat cu 0Bh;
DX - încărcat cu valoarea numărului de reîncercări;
CX - încărcat cu timpul de așteptare.

Dacă după retur este poziționat indicatorul CF, funcția returnează în AX codul de eroare 1 - funcție invalidă.

Observații:

1. Comanda SHARE este necesară înaintea apelului acestei subfuncții (altfel apare eroare 1 în AX).

2. Efectul parametrului din CX este dependent de mașină, întrucît specifică de cîte ori DOS va executa o buclă goală (implicit 1), ceea ce depinde de procesor și ceas.

3. Subfuncția este disponibilă doar în versiunile DOS >= 3.0.

4. Pentru informații suplimentare, a se vedea funcția 59h.

2.69.13. Funcția 440Ch - IOCTL: Cerere generică IOCTL pentru identificator logic

Subfuncția 0Ch poziționează sau returnează contorul de iterații la ieșire pentru o imprimantă care permite "Print til busy". Acest contor indică de cîte ori driverul de dispozitiv va aștepta semnalul "ready" de la dispozitiv, înainte de a semnala "Device Busy".

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 44h;
AL - încărcat cu 0Ch;
BX - încărcat cu identificatorul logic al dispozitivului;
CH - încărcat cu 05h (cod de categorie pentru dispozitiv imprimantă);
CL - încărcat cu:

45h - pentru poziționare;
65h - pentru returnare;

DS:DX - încărcat cu un pointer la o zonă tampon de date (cuvînt) care conține noua valoare a contorului.

Dacă după retur este poziționat indicatorul CF, funcția returnează în AX codul de eroare 1 - funcție invalidă.

Observație:

Subfuncția este disponibilă doar în versiunile DOS >= 3.2.



2.69.14. Funcţia 440Dh - IOCTL: Cerere generică IOCTL pentru dispozitive bloc

Subfuncţia 0Dh oferă un control de nivel scăzut asupra unui dispozitiv bloc.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 44h;
AL - încărcat cu 0Dh;
BL - încărcat cu numărul dispozitivului (0 = implicit, 1 = A, etc.);
CH - încărcat cu 08h (codul major - categorie).
CL - încărcat cu codul minor (subfuncţie):
40h - poziţionarea parametrilor dispozitivului;
60h - returnarea parametrilor dispozitivului;
41h - scrierea unei piste pe dispozitivul logic;
61h - citirea unei piste de pe dispozitivul logic;
42h - formatarea unei piste pe dispozitivul logic;
62h - verificarea unei piste depe dispozitivului logic.

Observaţii:

1. Pentru scrierea, citirea, formatarea sau verificarea unui dispozitiv logic, se recomandă procedura:

   a. Salvarea parametrilor dispozitivului cu subfuncţia 440D - 60h.
   b. Poziţionarea parametrilor doriţi cu subfuncţia 440D - 40h.
   c. Realizarea operaţiei de intrare/ieşire.
   d. Reîncărcarea parametrilor originali ai dispozitivului cu subfuncţia 440D - 40h.

2. Subfuncţia este valabilă doar în versiunile DOS >= 3.2.

2.69.14.1. Funcţia 440Dh (cod 40h) - Poziţionarea parametrilor dispozitivului

Dacă CL = 40h, blocul parametrilor are următorul format de cîmpuri:

Cîmp		Dimensiune
a.	Funcţii speciale	Octet
b.	Tipul dispozitivului	Octet
c.	Atributele dispozitivului	Cuvînt
d.	Numărul de unităţi de alocare ale discului	Cuvînt
e.	Caracteristicile dispozitivului	Octet
f.	BPB-ul dispozitivului	-
g.	Organizarea pistei	-

Cîmpurile au următoarele semnificaţii:

a. Funcţii speciale

Bit	Valoare	Semnificaţie
0	0	Cîmpul f conţine BPB-ul implicit al dispozitivului. Dacă un apel anterior (cod 40h) a poziţionat acest bit, o cerere BUILD BPB va returna BPB-ul actual, altfel BPB-ul implicit.
	1	Toate cererile BUILD BPB ulterioare returnează cîmpul f.
1	0	Citeşte toate cîmpurile din blocul de parametrii.
	1	Ignoră toate cîmpurile din blocul de parametrii, cu excepţia cîmpului g.
2	0	Sectoarele de pe pistă pot să nu aibă toate aceeaşi dimensiune (nu se recomandă utilizarea).
	1	Sectoarele de pe pistă au toate acceaşi dimensiune, iar numărul de ordine al sectorului este între 1 şi numărul total de sectoare aflate actual pe pistă. Se recomandă poziţionarea acestui bit.
3,4,5,6,7	0	Aceşti biţi trebuie să fie nuli.

b. Tipul dispozitivului

Acest octet descrie dispozitivul fizic şi este poziţionat de dispozitiv. Dacă este poziţionat, el are semnificaţiile:

Valoare	Semnificaţie
0	320/360 Kocteţi
1	1.2 Mocteţi
2	720 Kocteţi
3	8" simplă densitate
4	8" dublă densitate
5	Disc Winchester
6	Unitate de bandă
7	Alt tip

c. Atributele dispozitivului

Bit	Valoare	Semnificaţie
0	0	Se poate schimba.
0	1	Nu se poate schimba.
1	0	Nu permite detecţia schimbării dischetei.
1	1	Permite detecţia schimbării dischetei.
2,3,4,5,6,7	0	Aceşti biţi trebuie să fie nuli.

d. Numărul de unităţi de alocare ale discului

Acest cîmp indică numărul maxim de unităţi de alocare ale discului permise de dispozitivul fizic. Este poziţionat de dispozitiv.

e. Caracteristicile dispozitivului

Pentru unităţi care permit caracteristici diferite, acest cîmp (dependent de dispozitiv) indică ce caracteristici aşteaptă dispozitivul.

Pentru o dischetă de 1,2 Mocteţi, bitul zero are semnificaţiile:

Valoare	Semnificaţie
0	Mare capacitate, 1,2 Mocteţi (densitate cvadruplă)
1	Dublă densitate, 320/360 Kocteţi

Implicit este de mare capacitate, 1,2 Mocteţi.

f. BPB-ul dispozitivului

Dacă bitul zero al cîmpului a este nul, BPB-ul din acest cîmp este noul BPB implicit al dispozitivului. În caz contrar, driverul dispozitivului returnează BPB-ul din acest cîmp pentru apeluri BUILD BPB ulterioare.

g. Organizarea pistei

Cîmpul conţine un tabel cu lungimea variabilă a fiecarui dispozitiv logic şi indică organizarea aşteptată a sectoarelor pe pistă. Cîmpul are următorul format:

Numărul total de sectoare	Cuvînt
Număr sector (sector 1)	Cuvînt
Dimensiune sector (sector 1)	Cuvînt
Număr sector (sector 2)	Cuvînt
Dimensiune sector (sector 2)	Cuvînt
...	
...	
Număr sector (sector n)	Cuvînt
Dimensiune sector (sector n)	Cuvînt

Fiecare număr de sector trebuie să fie unic şi cuprins între 1 şi numărul total de sectoare. Dacă bitul doi din cîmpul a este poziţionat, toate sectoarele trebuie să aibă aceeaşi dimensiune.

2.69.14.2. Funcţia 440Dh (cod 60h) - Returnarea parametrilor dispozitivului

Dacă CL = 60h, blocul parametrilor are aceeaşi organizare ca la CL = 40h, dar unele cîmpuri au alte semnificaţii, şi anume:

a. Funcţii speciale

Bit	Valoare	Semnificaţie
0	0	Returnează BPB-ul implicit al dispozitivului.
	1	Returnează BPB-ul pe care l-ar returna apelul BUILD BPB.
1,2,3,4,5,6,7	0	Aceşti biţi trebuie să fie nuli.

g. Organizarea pistei

Acest cîmp nu este folosit.

2.69.14.3. Funcţia 440Dh (cod 41h/61h) - Scrierea/Citirea unei piste pe dispozitivul logic

Dacă CL=41h (scriere) sau CL=61h (citire), blocul parametrilor are următorul format:

Cîmp		Dimensiune
a.	Funcţii speciale	Octet
b.	Cap	Cuvînt
c.	Unitate de alocare a discului	Cuvînt
d.	Primul sector	Cuvînt
e.	Numărul de sectoare	Cuvînt
f.	Adresa de transfer	Dublu cuvînt

Cîmpurile au următoarele semnificaţii:

a. Funcţii speciale

Acest cîmp trebuie să fie nul.

b. Cap

Acest cîmp conţine numărul capului cu care se realizează scrierea/citirea.

c. Unitate de alocare a discului

Acest cîmp conţine numărul unităţii de alocare a discului pe care se realizează scrierea/citirea.

d. Primul sector

Acest cîmp conţine numărul primului sector pe/de pe care se realizează scrierea/citirea. Sectoarele se numerotează începînd cu zero.

e. Numărul de sectoare

Acest cîmp conţine numărul total de sectoare.

f. Adresa de transfer

Acest cîmp conţine adresa de stocare a datelor care vor fi scrise sau care vor fi citite.

2.69.14.4. Funcţia 440Dh (cod 42h/62h) - Formatarea/Verificarea unei piste pe dispozitivul logic

Dacă CL=42h (formatare) sau 62h (verificare), blocul parametrilor are următorul format:

Cîmp		Dimensiune
a.	Funcţii speciale	Octet
b.	Cap	Cuvînt
c.	Unitate de alocare a discului	Cuvînt

Cîmpurile au următoarele semnificaţii:

a. Funcţii speciale

Acest cîmp trebuie să fie nul.

b. Cap

Acest cîmp conţine numărul capului cu care se realizează formatarea/verificarea.

c. Unitate de alocare a discului

Acest cîmp conţine numărul unităţii de alocare a discului pe care se realizează formatarea/verificarea.

2.69.15. Funcţia 440Eh - IOCTL: Obţinerea dispozitivului logic

Anumite dispozitive bloc utilizează mai mult decît o singură literă de driver pentru a referi acelaşi dispozitiv (de exemplu, pe un sistem cu o singură unitate de dischetă, driverul de dischetă poate fi referit prin A: sau B:). Subfuncţia 0Eh indică dacă un acelaşi dispozitiv este referit prin mai mult de o liter de driver şi, dacă da, care literă de driver a fost utilizată cel mai recent.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 44h;
AL - încărcat cu 0Eh;
BX - încărcat cu numărul driverului (0 = implicit, 1 = A, etc).

Funcţia returnează următoarele:

- Dacă după retur este poziţionat indicatorul CF, un cod de eroare în AX:

1 - cod funcţie invalid;
5 - driver invalid.

- În caz contrar, 00h dacă dispozitivului bloc îi este ataşată doar o literă, respectiv litera de driver utilizată cel mai recent în caz contrar (1=A, 2=B, etc).



Observaţii:

1. Subfuncţia este disponibilă numai în versiunile DOS >= 3.2.

2. Pentru informaţii suplimentare, a se vedea funcţiile 44h, 440Fh.

2.69.16. Funcţia 440Fh - IOCTL: Selectarea dispozitivului logic

Subfuncţia 0Fh schimbă litera de driver cel mai recent utilizată pentru a referi un dispozitiv care poate fi specificat prin mai mult de o litera de driver.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 44h;
AL - încărcat cu 0Fh;
BX - încărcat cu numărul driverului (0 = implicit, 1 = A, etc).

Funcţia returnează următoarele:

- Dacă după retur este poziţionat indicatorul CF, un cod de eroare în AX:

1 - cod funcţie invalid;
5 - driver invalid.

- În caz contrar, 00h dacă dispozitivului bloc îi este ataşată doar o literă, respectiv cea mai "înaltă" literă asignată dispozitivului bloc (1 = A, 2 = B, etc).

Observaţii:

1. O operaţie de intrare/ieşire care utilizează o liter de driver diferită de cea mai recent utilizată va duce la generarea unui mesaj "Insert diskette for ... ". Pentru a-l evita, trebuie schimbată litera curentă înaintea efecturii operaţiei.

2. Subfuncţia este disponibilă doar în versiunile DOS >= 3.2.

3. Pentru informaţii suplimentare, a se vedea funcţiile 44h, 440Eh.

2.70. Funcţia 45h - Duplicarea identificatorului logic al unui fişier (DUP)

Funcţia 45h creează un identificator logic de fişier adiţional pentru un fişier existent. Noul identificator logic de fişier se referă la acelaşi fişier (specificat), cu pointerul de scriere/ieşire în aceeaşi poziţie. În continuare, orice acţiune asupra pointerului unuia dintre fişiere va deplasa în mod identic pointerul celuilalt fişier. Modul de deschidere al fişierului copie este identic cu cel al fişierului original.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 45h;
BX - încărcat cu identificatorul logic al unui fişier deschis.

Dacă după retur nu este poziţionat indicatorul CF, funcţia va returna în AX noul identificator logic de fişier, respectiv un cod de eroare în caz contrar:

4 - prea multe fişiere deschise;
6 - identificator logic de fişier nedeschis sau invalid.

Observaţii:

1. Închiderea unuia dintre fişiere nu îl va închide şi pe al doilea. Se poate deci închide unul dintre fişiere actualizînd directorul, în timp ce al doilea identificator logic de fişier rămîne deschis. Secvenţa duplicare/închidere este mai rapidă decît închiderea/deschiderea fişierului.

2. Funcţia este în general utilizată pentru redirectarea intrării standard (identificator logic 0) şi/sau ieşirii standard (identificator logic 1).

3. Pentru informaţii suplimentare, a se vedea funcţiile 46h, 59h.

2.71. Funcţia 46h - Redirectarea unui identificator logic de fişier (FORCDUP)

Funcţia 46h forţează un identificator logic de fişier (specificat în CX) să refere acelaşi fişier ca şi un al doilea identificator logic de fişier (specificat în BX), acesta din urmă fiind deja asociat unui fişier deschis. Ambele identificatoare logice de fişier trebuie să existe. Dacă primul identificator logic de fişier se referă la un fişier deschis, acesta va fi închis înaintea efectuarii redirectării. Pointerul de scriere/citire va fi în aceeaşi poziţie, iar orice acţiune asupra pointerului unuia dintre fişiere va deplasa în mod identic pointerul celuilalt fişier. Modul de deschidere al celui de al doilea identificator logic de fişier va fi identic cu al primului identificator logic de fişier.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 46h;
BX - încărcat cu primul identificator logic de fişier;
CX - încărcat cu al doilea identificator logic de fişier.

Dacă după retur este poziţionat indicatorul CF, funcţia va returna în AX un cod de eroare:

4- prea multe fişiere deschise;
6- identificator logic de fişier nedeschis sau invalid.

Observaţii:

1. Închiderea unuia dintre identificatoarele logice de fişier nu îl va închide şi pe celălalt.

2. Funcţia este în general utilizată pentru redirectarea intrării standard (identificator logic 0) şi/sau ieşirii standard (identificator logic 1).

3. Pentru informaţii suplimentare, a se vedea funcţiile 45h, 59h.

2.72. Funcţia 47h - Obţinerea directorului curent

Funcţia 47h returnează numele de cale pentru directorul curent al dispozitivului specificat. În zona de memorie avînd specificat în SI deplasamentul faţă de adresa segment din DS va fi plasat un şir ASCIIZ în format "nume-cale\nume-director", 0. Litera pentru dispozitiv şi carcaterul 'backslash' iniţial lipsesc.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 47h;
DS:SI - încărcat cu un pointer la o zonă de memorie de 64 octeţi;
DL - încărcat cu numărul dispozitivului.

Dacă este poziţionat indicatorul CF, funcţia va returna în AX codul de eroare 15 - număr dispozitiv invalid.

Observaţii:

1. Dacă directorul curent este rădăcină, se va returna un şir nul.

2. În versiunile DOS anterioare lui 3.0, numele de cale poate conţine membrii avînd chiar mai mult de 12 caractere, iar numele de cale va fi returnat cu litere mici sau mari (după caz). În versiunile DOS >= 3.0, numele membrilor este trunchiat la 8 caractere, iar returnarea se face doar cu litere mari.

3. Pentru informaţii suplimentare, a se vedea funcţiile 39h, 3Ah, 3Bh, 59h.

2.73. Funcţia 48h - Alocarea memoriei

Funcţia 48h încearcă să aloce dinamic procesului curent numărul specificat de paragrafe de memorie (un paragraf = 16 octeţi).

Pentru apelul funcţiei, se folosesc regiştrii:

AX - încărcat cu 48h;

BX -încărcat cu numărul de paragrafe de memorie.

Funcția returnează următoarele:

- Dacă este poziționat indicatorul CF, un cod de eroare în AX:

  7 - deteriorarea blocurilor de control ale memoriei (un program utilizator a modificat memoria care nu îi aparține);
  8 - memorie insuficientă;

  și dacă nu există suficientă memorie, numărul paragrafelor de memorie din cel mai mare bloc disponibil, în BX.

- Dacă alocarea a reușit, adresa segment a memoriei alocate, în AX (deplasament 0000h).

Observații:

1. Cînd un proces primește controlul prin funcția 4Bh, toată memoria disponibilă i-a fost deja alocată.

2. Pentru a determina cel mai mare bloc de memorie disponibil, se poate apela funcția cu BX = FFFFh. Deoarece sistemul de operare nu poate aloca mai mult de 640 Kocteți de memorie, va rezulta o eroare care se va ignora, registrul BX fiind poziționat în sensul dorit.

3. Pentru informații suplimentare, a se vedea funcțiile 49h, 4Ah, 4Bh, 59h.

2.74. Functi7a 49h - Eliberarea memoriei alocate

Funcția 49h eliberează (făcîndu-l disponibil pentru sistem) un bloc de memorie alocat anterior prin funcția 48h.

Pentru apelul funcției, se folosesc regiștrii:

AH -încărcat cu 49h;

ES - încărcat cu adresa segment a memoriei de eliberat (lungimea fiind obţinută de sistem din blocul de control din faţa zonei de memorie).

Dacă după retur este poziţionat indicatorul CF, funcţia va returna un cod de eroare în AX:

7 - deteriorarea blocurilor de control ale memoriei (un program utilizator a modificat memoria care nu îi aparţine).
9 - segment incorect (memoria pointată de ES nu a fost alocată cu funcţia 48h).

Observaţii:

1. Procesul părinte trebuie să ţină evidenţa memoriei alocate procesului fiu. Este posibilă şi dealocarea memoriei altor procese.

2. Pentru informaţii suplimentare, a se vedea funcţiile 48h, 4Ah, 4Bh 59h.

2.75. Funcţia 4Ah - Modificarea alocării memoriei (SET BLOCK)

Funcţia 4Ah încearcă să modifice dimensiunea unui bloc de memorie alocat prin funcţia 48h.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 4Ah;
BX - încărcat cu noua dimensiune a memoriei în număr de paragrafe (un paragraf = 16 octeţi);
ES - încărcat cu adresa segment a zonei de memorie.

Dacă după retur este poziţionat indicatorul CF, funcţia va returna în BX dimensiunea maximă (în paragrafe) la care poate fi mărită memoria (dacă s-a dorit mărirea), şi un cod de eroare în AX:

7 - deteriorarea blocurilor de control ale memoriei (un program utilizator a modificat memoria care nu îi aparţine);
8 - memorie insuficientă;
9 - adresa eronată în ES (blocul de memorie pointat nu poate fi modificat cu funcţia 4Ah).

Observaţii:

1. Cînd un proces primeşte controlul prin funcţia 4Bh, toată memoria disponibilă i-a fost deja alocată. Eliberarea de memorie pentru a lansa un proces fiu necesită întîi o reducere a blocului de memorie care începe de la PSP.

2. Pentru informaţii suplimentare, a se vedea funcţiile 48h, 49h, 4Ah, 4Bh, 59h.

2.76. Funcţia 4Bh- Încărcarea şi/sau execuţia unui program (EXEC)

Funcţia 4Bh permite unui program să încarce în memorie un subprogram (al cărui nume este specificat printr-un şir ASCIIZ "d:\nume_cale\specif_fişier", 0 avînd deplasamentul faţă de adresa segment din DS în registrul DX) şi, opţional, să îl execute.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 4Bh;
AL - încărcat cu 00h, pentru încărcare şi execuţie;
- încărcat cu 03h, pentru încărcare fără execuţie (cazul tipic fiind încărcarea segmentelor de program);
DS:DX - încărcat cu un pointer la numele subprogramului şir ASCIIZ);
ES:BX - încărcat cu un pointer la un bloc de parametrii.

Dacă după retur este poziţionat indicatorul CF, funcţia returnează în AX un cod de eroare:

1 - funcţie invalidă (AL nu este 0 sau 3);
2 - nu s-a găsit fişierul;
3 - nu s-a găsit calea;
4 - prea multe fişiere deschise;
5 - acces interzis;
8 - memorie insuficientă;
10 - mediu eronat (peste 32 Kocteţi);
11 - format eronat (fişier .EXE cu informaţii interne inconsistente).

Blocul de parametri are următorul format:

. Pentru AL = 00h:

Deplasament	Lungime (octeţi)	Descriere
00h	2(cuvînt)	Adresa de segment a şirurilor de mediu (maximum 32 Kocteţi), valoarea 0 însemnînd copierea mediului părintelui. Se va plasa la deplasament 2Ch în noul PSP.
02h	4(dublu cuvînt)	Adresa de segment şi deplasamentul liniei de comandă (maximum 128 octeţi). Se va plasa la deplasament 80h în noul PSP.
06h	4(dublu cuvînt)	Adresa de segment şi deplasamentul primului FCB implicit. Se va plasa la deplasament 5Ch în noul PSP.
0Ah	4(dublu cuvînt)	Adresa de segment şi deplasamentul celui de-al doilea FCB implicit. Se va plasa la deplasament 6Ch în noul PSP.

. Pentru AL = 03h:

Deplasament	Lungime (octeți)	Descriere
00h	2(cuvînt)	Adresa de segment unde va fi încărcat programul.
02h	2(cuvînt)	Factorul de relocare al programului (doar pentru fișiere .EXE).

În cazul execuției, DOS creează un nou PSP pentru programul încărcat. Vectorii INT 22h, INT 23h și INT 24h sînt poziționați la instrucțiunea ce urmează imediat după apelul funcției EXEC, astfel încît controlul revine programului apelant imediat după terminarea subprogramului. Întrucît DOS alocă toată memoria disponibilă unui program pe care îl încarcă, înainte de apel trebuie eliberată o porțiune de memorie (funcția 4Ah) pentru ca subprogramul să poată fi încărcat și executat. Toate identificatoarele logice de fișiere sînt disponibile subprogramului, astfel încît programul apelant are controlul asupra definirii intrărilor/ieșirilor standard și auxiliare și a dispozitivelor de tipărire. Singurele identificatoare logice indisponibile procesului fiu sînt cele deschise cu bitul de moștenire poziționat pe 1 (funcția 3Dh).

Șirurile de mediu (care trebuie să fie aliniate la granița de paragraf) reprezintă o suită de șiruri ASCIIZ, urmate fiecare de un octet nul, lista de șiruri fiind terminată tot cu un octet nul (ultimul șir va fi deci urmat de doi octeți nuli). Fiecare șir este de forma:

parametru = valoare

de exemplu, VERIFY = ON.

Dacă adresa de segment a șirurilor de mediu este 00h, subprogramul moștenește mediul procesului părinte. Subprogramul primește întotdeauna o copie a mediului, astfel încît schimbările în mediul său nu se vor reflecta în mediul programului apelant.

În cazul în care subprogramul este încărcat fără execuție (existînd posibilitatea de a se face salt la el), nu este necesară eliberarea de memorie și nu se crează PSP.

Observații:

1. Cea mai simplă metodă de a încărca și executa un alt program constă în încărcarea și execuția unei copii adiționale a lui COMMAND.COM, folosind o linie de comandă cu comutatorul /C, care apelează fișierul .COM sau .EXE (operațiunea necesită 17 Kocteți de memorie).

   Linia de comandă destinată interpretorului de comenzi secundar va avea formatul:

   <lungime>/C <comanda> <CR>

unde:

lungime - reprezintă lungimea liniei de comandă (inclusiv octetul de lungime, dar fără <CR>);
comanda - reprezintă orice comandă DOS validă.

   În acest caz, registrii DS:DX trebuie să conțină adresa șirului "COMMAND.COM", iar blocul de parametri (deplasament 2) adresa liniei de comandă specificată mai sus.

   Dacă un program execută un alt program direct, ei trebuie să realizeze toate prelucrările făcute în mod normal de COMMAND.COM.

2. Pentru informații suplimentare, a se vedea funcțiile 3Dh, 4Ah, 4Dh, 59h, FCB și PSP.

2.77. Funcția 4Ch - Terminarea procesului (EXIT)

Funcția 4Ch termină un program și predă controlul procesului părinte, poziționînd un cod de retur care poate fi prelucrat de către procesul apelant prin funcția DOS 4Dh sau comanda IF utilizînd ERRORLEVEL. Toate fișierele sînt închise,

iar vectorii Ctrl/Break (INT 23h) şi Eroare Critica (INT 24h) sînt reîncărcaţi la adresa din PSP-ul părintelui.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 4Ch;
AL - încărcat cu codul de retur.

Funcţia nu returnează nimic.

Observaţii:

1. Se recomandă utilizarea acestei funcţii în locul lui INT 20h, dacă programul nu trebuie să fie compatibil cu versiuni anterioare lui DOS 2.0.

2. Dacă se utilizează partajarea fişierelor, toate blocările introduse de proces trebuie eliminate, altfel DOS va ajunge într-o stare necunoscută.

3. Pentru informaţii suplimentare, a se vedea funcţiile 00h, 31h, 4Dh şi INT 20h, INT 27h.

2.78. Funcţia 4Dh - Obţinerea codului de retur al procesului fiu (WAIT)

Funcţia 4Dh returnează codul de ieşire al unui proces terminat prin una din funcţiile 4Ch sau 31h.

Pentru apelul funcţiei, se foloseşte registrul:

AH - încărcat cu 4Dh.

Funcţia returnează în AX codul de retur.

Observaţii:

1. Funcţia returnează informaţii valide doar o singură dată pentru fiecare proces terminat.

2. Informaţia returnată constă din două părţi:

AL - conţine codul returnat de către însuşi procesul terminat;
AH - conţine un cod care specifică în ce mod s-a terminat procesul:

Cod	Semnificaţie
0	Terminare normală.
1	Terminare prin Ctrl\Break sau Ctrl\C.
2	Terminare de către DOS datorită unei erori critice de dispozitiv.
3	Terminare prin funcţia 31h.

2.79. Funcţia 4Eh - Găsirea primului fişier potrivit (FIND FIRST)

Funcţia 4Eh caută prima intrare în director potrivită cu cea specificată sub forma unui şir ASCIIZ ("d:\nume_cale\specif_fişier:",0), al cărui deplasament faţă de adresa segment din DS se află în DX. Dacă dispozitivul şi calea sînt omise, se vor considera cele implicite. Numele de fişier poate conţine caracterele "?" şi/sau "*".

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 4Eh;
DS:DX - încărcat cu un pointer la numele de cale;
CX - încărcat cu atributele fişierului căutat.

Dacă apare o eroare, este poziţionat indicatorul CF şi se returnează în AX un cod de eroare:

2 - nu s-a găsit fişierul;
3 - nu s-a găsit calea;
18 - nu mai există fişiere.

Dacă se găseşte o intrare în director potrivită, funcţia returnează la DTA curentă informaţii despre fişiere, în următorul format:

Deplasament	Lungime	Descriere
00h	21	Rezervați pentru apeluri ulterioare ale funcției DOS 4Fh.
15h	1	Atributul fișierului găsit.
16h	2	Ora ultimei modificări a fișierului găsit.
18h	2	Data ultimei modificări a fișierului găsit.
1Ah	4	Dimensiunea fișierului găsit (în octeți).
1Eh	13	Numele și extensia fișierului, ca șir ASCIIZ.

Observații:

1. Spațiile din nume și/sau extensie sînt eliminate. Dacă există extensie, ea este precedată de caracterul "." (dacă nu există extensie, caracterul "." este eliminat). Etichetele de volum includ caracterul "." după al optulea caracter.

2. Sfera de căutare poate fi controlată, utilizînd octetul de atribut din CX. Dacă CX este 0, funcția caută doar fișierele normale. Dacă în CX este specificată o combinație de biți pentru alte tipuri de fișiere (ascunse, sistem, director, eticheta de volum), vor fi căutate acestea, împreună cu cele normale. Biții de atribut arhivă și "read-only" nu au efect în căutare.

3. Pentru informații suplimentare, a se vedea funcțiile 1Ah, 4Fh, 11h, 12h, 59h.

2.80. Funcția 4Fh - Găsirea următorului fișier potrivit (FIND NEXT)

Funcția 4Fh caută următoarea intrare în director potrivită cu fișierul specificat într-un apel precedent al funcției 4Eh. Funcția 4Fh se va utiliza după un apel precedent al funcției 4Eh sau/și 4Fh.

Pentru apelul funcției, se folosește registrul:

AH - încărcat cu 4Fh.

Dacă după retur este poziţionat indicatorul CF, funcţia returnează în AX codul de eroare 18 - nu mai există fişiere, respectiv informaţii despre fişierul găsit în caz contrar. Informaţia este pusă la DTA curentă, avînd aceeaşi structură ca la funcţia DOS 4Eh.

Pentru informaţii suplimentare, a se vedea funcţiile 1Ah, 4Eh, 11h, 12h, 59h.

2.81. Funcţia 50h - Rezervată

2.82. Funcţia 51h - Rezervată

2.83. Funcţia 52h - Rezervată

2.84. Funcţia 53h - Rezervată

2.85. Funcţia 54h - Verificarea stării indicatorului VERIFY

Funcţia 54h cercetează dacă DOS verifică sau nu operaţiile de scriere în fişiere disc.

Pentru apelul funcţiei, se foloseşte registrul:

AH - încărcat cu 54h.

Funcţia returnează în AL starea indicatorului VERIFY:

0 - VERIFY OFF (nu se verifică scrierea);
1 - VERIFY ON (se verifică scrierea).

Starea indicatorului poate fi modificată cu funcţia 2Eh sau prin comanda DOS VERIFY.

2.86. Funcţia 55h - Rezervată

2.87. Funcţia 56h - Schimbarea intrării în director

Funcţia 56h redenumeşte un fişier existent şi/sau îl mută într-un alt director de pe acelaşi disc. Numele de fişiere sînt specificate sub formă de şiruri ASCIIZ ("d:\nume_cale\specif_fişier",0) şi nu pot conţine caracterele "?" şi "*". Dacă dispozitivul şi numele de cale sînt omise, se vor lua cele implicite.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 56h;
DS:DX - încărcat cu un pointer la un şir ASCIIZ conţinînd vechea intrare în director;
ES:DI - încărcat cu un pointer la un şir ASCIIZ conţinînd noua intrare în director.

Dacă după retur este poziţionat indicatorul CF, funcţia returnează în AX un cod de eroare:

2 - nu s-a găsit fişierul;
3 - nu s-a găsit calea;
5 - acces interzis;
17 - dispozitive diferite.

Observaţii:

1. Funcţia nu poate fi utilizată pentru copierea fişierelor.

2. Fişierele deschise, ascunse, sistem şi subdirectoare nu pot fi redenumite.

3. Pentru informaţii suplimentare, a se vedea funcţiile 17h, 59h.

2.88. Funcţia 57h - Obţinerea/Poziţionarea datei şi orei unui fişier

Funcţia 57h returnează sau poziţionează data şi ora ultimei modificări a unui fişier deschis printr-un identificator logic.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 57h;
BX - încărcat cu identificatorul logic al fişierului;
AL - încărcat cu 00h pentru obţinerea datei şi orei;
- încărcat cu 01h pentru poziţionarea datei şi orei;
CX - încărcat cu ora de poziţionat, dacă AL = 01h;
DX - încărcat cu data de poziţionat, dacă AL = 01h.

Dacă după retur este poziţionat indicatorul CF, funcţia returnează în AX un cod de eroare:

1 - funcţie invalidă (AL nu este 0 sau 1);
6 - identificator logic invalid sau nedeschis.

În caz contrar şi dacă AL = 00h, funcţia returnează în CX ora şi în DX data.

Observaţii:

1. Data şi ora sînt definite astfel:

CX = Ora*2048 + Minut*32 + Secunda/2
DX = (An-1980)*512 + Luna*32 + Zi

Biţi CX	Descriere
11-15	Ora (0...23)
5-10	Minut (0...59)
0-4	Secunda (0...30)

Biţi DX	Descriere
9-15	An după 1980 (0...119)
5-8	Luna (1...12)
0-4	Zi (1...31)

2. Data şi ora sînt returnate în formatul convenţional, cu partea cea mai semnificativă în DH, respectiv CH.

3. Pentru informaţii suplimentare, a se vedea funcţiile 3Dh, 59h.

2.89. Funcția 58h - Obținerea/Selectarea strategiei de alocare

Funcția 58h returnează sau poziționează strategia utilizată de DOS pentru a aloca memoria, atunci cînd un proces o cere.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 58h;
AL - încărcat cu 00h pentru obținerea strategiei;
- încărcat cu 01h pentru selectarea strategiei;
BX - încărcat cu tipul strategiei, dacă AL = 01h:
0 - First Fit;
1 - Best Fit;
2 - Last Fit.

Dacă după retur este poziționat indicatorul CF, funcția returnează în AX codul de eroare 1, specificînd funcție invalidă (AL nu este 0 sau 1, sau BX nu este 0, 1 sau 2).

În caz contrar și dacă AL = 00h, funcția returnează în AX tipul strategiei de alocare utilizate:

0 - First Fit;
1 - Best Fit;
2 - Last Fit.

Cele trei strategii posibile sînt:

Valoare	Nume	Descriere
0	First Fit	DOS începe căutarea de la (Prima Potrivire) blocul de memorie disponibil de la cea mai joasă adresă și alocă primul bloc găsit (memoria alocată este blocul disponibil de la cea mai joasă adresă). Aceasta este strategia implicită.
1	Best Fit	DOS verifică fiecare bloc de (Cea mai bună memorie disponibil și alocă cel

		potrivire) mai mic bloc disponibil care satisface cererea.
2	Last Fit	DOS începe căutarea de la (Ultima potrivire) blocul de memorie disponibil de la cea mai înaltă adresă și alocă primul bloc găsit (memoria alocată este blocul liber de la cea mai înaltă adresă).

2.90 Funcția 59h - Obținerea informațiilor suplimentare despre o eroare

Funcția 59h returnează informații de diagnostic despre o condiție de eroare apărută în apelul sistem precedent. Funcția poate fi utilizată din interiorul unei rutine de tratare de eroare critică (INT 24h), după un apel de funcție DOS care semnalizează o eroare poziționînd indicatorul CF (Carry Flag), sau după o operație cu un fișier FCB care returnează un cod FFh.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 59h;
BX - încărcat cu 00h pentru versiuni DOS >= 3.0 (indică versiunea funcției, fiind necesar aplicațiilor rulate sub versiuni noi ale sistemului).

Funcția returnează următoarele:

. în AX - codul extins al erorii;
. în BX - clasa erorii;
. în BL - acțiunea sugerată;
. în CH - sursa erorii.

Codul extins al erorii:

Valoarea AX	Descriere
01h	Număr funcție invalid.
02h	Fişier negăsit.
03h	Cale negăsită.
04h	Prea multe fişiere deschise.
05h	Acces interzis.
06h	Identificator logic de fişier invalid.
07h	Blocurile de control ale memoriei distruse.
08h	Memorie insuficientă.
09h	Adresa invalidă a blocului de memorie.
0Ah	Mediu invalid.
0Bh	Format invalid.
0Ch	Cod de acces invalid.
0Dh	Data invalidă.
0Fh	Driver de disc invalid.
10h	Încercare de a şterge directorul curent.
11h	Dispozitive diferite.
12h	Nu mai există fişiere.
13h	Disc protejat la scriere.
14h	Unitate necunoscută.
15h	Driver nepregătit.
16h	Comanda necunoscută.
17h	Eroare de dată (CRC).
18h	Cerere greşită pentru lungimea structurii.
19h	Eroare în căutare.
1Ah	Tip unitate necunoscut.
1Bh	Sector negăsit.
1Ch	Lipsă de hirtie la imprimantă.
1Dh	Eroare în scriere.
1Eh	Eroare de citire.
1Fh	Eşec general.
20h	Violarea partajării fişierelor.
21h	Violarea unui blocaj.
22h	Schimbare de disc invalidă.
23h	FCB indisponibil.
50h	Fişier existent.
52h	Nu se poate crea directorul.
53h	Eroare critică.

Clase de eroare:

Valoare BH	Descriere
01h	Lipsă de resurse.
02h	Nu e eroare, ci o problemă temporară, căreia i se aşteaptă terminarea.
03h	Problemă de autorizare (permisie).
04h	Eroare internă în soft-ul sistemului.
05h	Eroare hard.
06h	Eşec în software-ul sistemului, nedatorat programului curent (de exemplu, fişier CONFIG.SYS lipsă sau incorect).
07h	Eroare a programului de aplicaţie.
08h	Fişier sau alt obiect negăsit.
09h	Fişier sau alt obiect în format sau tip invalid sau nepotrivit.
0Ah	Fişier sau alt obiect interblocat.
0Bh	Problemă cu tipul de disc (de exemplu, disc greşit în unitate).
0Ch	Contradicţie cu un obiect existent.
0Dh	Altă eroare.

Acţiune sugerată:

Valoare BL	Descriere
01h	Reluare.
02h	Reluare, după o pauză.
03h	Cerere pentru utilizator de a reintroduce data.
04h	Terminare cu curăţire.
05h	Terminare imediată (sistem total defectuos, programul trebuie terminat rapid, fără a se mai face actualizări şi inchideri de fişiere).
06h	Eroare informaţională (se ignoră).
07h	Cere utilizatorului să facă o acţiune (de exemplu, să insereze o dischetă), apoi se reia.

Sursa erorii:

Valoare CH	Descriere
01h	Necunoscută.
02h	Legată de un dispozitiv bloc (de exemplu, disc).
03h	Legată de rețea.
04h	Legată de un dispozitiv serial (de exemplu, imprimantă).
05h	Legată de memoria RAM.

Observații:

1. Funcția este disponibilă numai în versiunile DOS >= 3.0. Codul erorilor se va extinde la fiecare nouă versiune DOS.

2. Pentru informații suplimentare, a se vedea INT 24h.

ATENȚIE!

Funcția distruge regiștrii CL, DX, SI, DI, ES, DS.

2.91. Funcția 5Ah - Crearea unui fișier temporar unic

Funcția 5Ah crează un fișier cu un nume unic. Numele fișierului creat este adăugat la numele de cale specificat sub forma unui șir ASCIIZ ("d:\nume_cale\",0) avînd deplasamentul față de adresa segment din DS în registrul DX. În acest scop, după numele de cale trebuie rezervați 13 octeți de memorie pentru numele fișierului. Fișierul este deschis în mod compatibilitate. Dacă numele dispozitivului și al căii sînt omise, se vor lua cele implicite.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 5Ah;
CX - încărcat cu atributul fișierului;
DS:DX - încărcat cu un pointer la un nume de cale (șir ASCIIZ) terminat cu caracterul "\" și urmat de 13 octeți rezervați.

Dacă după retur nu este poziţionat indicatorul CF, funcţia returnează în AX identificator logic al fişierului, respectiv un cod de eroare în caz contrar:

3 - nu s-a găsit calea;
4 - prea multe fişiere deschise;
5 - acces interzis.

Observaţii:

1. La terminarea programului, fişierul nu este şters automat de către DOS.

2. Numele fişierului este construit de DOS din cifre hexa obţinute din data şi ora curentă. Dacă numele de fişier există deja, DOS reia operaţiunea pînă cînd numele de fişier poate fi creat.

3. Funcţia este disponibilă numai în versiunile DOS >= 3.0.

4. Pentru informaţii suplimentare, a se vedea funcţiile 16h, 3Ch, 5Bh, 59h.

2.92. Funcţia 5Bh - Crearea unui fişier nou

Funcţia 5Bh creează un nou fişier (dacă şi numai dacă nu există deja un fişier cu acelaşi nume), utilizînd un şir ASCIIZ ("d\nume_cale\specif_fişier\",0) al cărui deplasament faţă de adresa registru din DS se află în DX. Fişierul este deschis în mod compatibilitate. Dacă numele dispozitivului şi/sau al căii este omis, se vor lua cele implicite.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 5Bh;
CX - încărcat cu atributele fişierului;
DS:DX - încărcat cu un pointer la un nume de cale (şir ASCIIZ).

Dacă după retur nu este poziţionat indicatorul CF, funcţia returnează în AX identificatorul logic al fişierului, respectiv un cod de eroare, în caz contrar:

3 - nu s-a găsit calea;
4 - prea multe fişiere deschise;
5 - acces interzis;
80 - fişier existent.

Observaţii:

1. Funcţia este identică cu funcţia 3Ch, dacă fişierul specificat nu există. În sisteme multitasking, existenţa fişierului poate fi folosită ca un semafor.

2. Funcţia este disponibilă doar în versiunile DOS >= 3.0.

3. Pentru informaţii suplimentare, a se vedea funcţiile 16h, 3Ch, 5Ah, 59h.

2.93. Funcţia 5Ch - Blocare/Deblocarea accesului la un fişier

Funcţia 5Ch permite sau împiedică accesul la o regiune specificată unui fişier, într-un mediu multitasking sau reţea.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 5Ch;
AL - încărcat cu 00h, pentru blocare;
- încărcat cu 01h, pentru deblocare;
BX - încărcat cu identificatorul logic al fişierului;
CX:DX - încărcat cu deplasamentul regiunii în fişier;
SI:DI - încărcat cu lungimea regiunii.

Dacă după retur este poziţionat indicatorul CF, funcţia returnează în AX un cod de eroare:

1 - cod funcţie invalid;
6 - identificator logic invalid sau nedeschis;
33 - blocaj violat (regiunea este deja blocată);

36 - zonă tampon cu intrări de blocaj depăşită.

Observaţii:

1. Funcţia necesită instalarea partajării fişierelor prin comanda DOS SHARE.

2. Blocarea se aplică pentru scrierea, citirea şi deschiderea fişierului de către procese fiu sau concurente. O tentativă de acest gen va fi încercată de 3 ori de DOS, cu afişarea mesajului: "Abort, Retry, Ignore". După al treilea eşec, se va genera INT 24h pentru procesul respectiv. Se recomandă iniţial testarea stării regiunii, printr-o încercare de a o bloca.

3. O regiune trebuie să fie blocată pentru cît mai puţin timp posibil. Regiunea trebuie deblocată înaintea închiderii fişierului, iar programele care pot fi terminate prin INT 23h sau INT 24h trebuie să capteze aceste întreruperi şi să deblocheze regiunile blocate înainte de ieşire.

4. La deblocare, deplasamentul şi lungimea regiunii trebuie să fie identice cu cele din cazul blocării.

5. Multiplicarea unui fişier prin funcţiile 45h sau 46h va multiplica şi toate blocările.

6. Un proces fiu creat prin funcţia 4Bh nu moşteneşte drepturile de acces ale părintelui.

7. Orice regiune a unui fişier poate fi blocată. Poate fi blocat tot fişierul (CX = 0, DX = 0, SI = 0FFFFh, DI = 0FFFFh, AL = 0), iar blocajul după EOF nu constituie o eroare.

8. Funcţia este disponibilă doar în versiunile DOS >= 3.0.

9. Pentru informaţii suplimentare, a se vedea funcţiile 3Dh, 59h, 440Bh şi comanda SHARE.

ATENȚIE!

Dacă un program închide un fișier care conține o regiune blocată sau se termină cu un fișier deschis avînd o regiune blocată, rezultatul este nedefinit.

2.94. Funcția 5Dh - Rezervată

2.95. Funcția 5Eh - Servicii de rețea

Funcția 5Eh este disponibilă începînd cu versiunea DOS 3.1, dacă este încărcat Programul de Rețea IBM PC sau Microsoft.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 5Eh;
AL - încărcat cu codul subfuncției.

Subfuncțiile disponibile sînt:

00h - obținerea numelui mașinii;
02h - poziționarea antetului imprimantei de rețea;
03h - obținerea antetului imprimantei de rețea.

2.95.1. Funcția 5E00h - Obținerea numelui mașinii

Subfuncția 00h returnează numele calculatorului local sub forma unui șir ASCIIZ, plasîndu-l într-o zonă tampon de 16 octeți al cărei deplasament față de adresa segment din DS se află în DX. La nevoie, numele este completat cu blancuri.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 5Eh;
AL - încărcat cu 00h;
DS:DX - încărcat cu un pointer la o zona tampon de memorie de 16 octeți.

Funcţia returnează următoarele:

. în CH - 00h, dacă numele nu a fost definit;
. în CL - numărul numelui NETBIOS, dacă CH nu e zero;
- nedefinit, dacă CH e zero;
. în AX - dacă indicatorul CF este poziţionat, codul de eroare 1 - nu a fost încărcat programul de reţea.

Observaţii:

1. Subfuncţia este disponibilă doar în versiunile DOS >= 3.1, şi necesită încărcarea programului de reţea IBM PC sau Microsoft.

2. Pentru informaţii suplimentare, a se vedea funcţia 59h.

2.95.2. Funcţia 5E02h - Poziţionarea antetului imprimantei de reţea

Subfuncţia 02h specifică şirul de antet care va fi pus la începutul fiecărui fişier destinat unei imprimante de reţea. Astfel, fiecare utilizator individual îşi poate specifica un antet pentru o imprimantă partajată.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 5Eh;
AL - încărcat cu 02h;
BX - încărcat cu indexul listei de redirectare;
CX - încărcat cu lungimea şirului de antet (maximum 64 octeţi);
DS:SI - încărcat cu un pointer la o zonă tampon de memorie conţinînd şirul de antet.

Dacă după retur este poziţionat indicatorul CF, funcţia returnează în AX codul de eroare 1 - nu a fost încărcat programul de reţea.

Observaţii:

1. Subfuncţia este disponibilă doar în versiunile DOS >= 3.1, şi necesită încărcarea programului de reţea IBM PC sau Microsoft.

2. Valoarea de index din BX poate fi determinată cu funcţia 5F02h. Întrucît redirectarea se poate modifica în timp, se recomandă apelul consecutiv al funcţiilor 5F02h şi 5E02h.

3. Pentru informaţii suplimentare, a se vedea funcţiile 5E03h, 5F02h, 5F03h, 5F04h, 59h.

2.95.3. Funcţia 5E03h - Obţinerea antetului imprimantei de reţea

Subfuncţia 03h returnează şirul de antet care va fi plasat la începutul fiecărui fişier destinat unei imprimante de reţea, plasîndu-l într-o zonă tampon de memorie al cărei deplasament faţă de adresa segment din DS se află în DX.

Pentru apelul funcţiei, se folosesc regiştrii:

AH	- încărcat cu 5Eh;
AL	- încărcat cu 03h;
BX	- încărcat cu indexul listei de redirectare;
ES:DI	- încărcat cu un pointer la o zonă tampon de memorie de 64 octeţi.

Dacă după retur nu este poziţionat indicatorul CF, funcţia returnează în CX lungimea şirului de antet (maximum 64 octeţi), respectiv în AX codul de eroare 1 - nu s-a încărcat programul de reţea, în caz contrar.

Observaţii:

1. Subfuncţia este disponibilă doar în versiunile DOS >= 3.1, şi necesită încărcarea programului de reţea IBM PC sau Microsoft.

2. Valoarea de index din BX poate fi determinată cu funcția 5F02h. Întrucît redirectarea se poate modifica în timp, se recomandă apelul consecutiv al funcțiilor 5F02h și 5E02h.

3. Pentru informații suplimentare, a se vedea funcțiile 5E02h, 5F02h, 5F03h, 5F04h, 59h.

2.96. Funcția 5Fh - Redirectarea dispozitivelor de rețea

Funcția 5Fh este disponibilă începînd cu versiunea DOS 3.1, dacă este încărcat Programul de Rețea IBM PC sau Microsoft.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 5Fh;
AL - încărcat cu codul subfuncției.

Subfuncțiile disponibile sînt:

02h - obținerea unei intrări în lista de redirectări;
03h - redirectarea unui dispozitiv de rețea;
04h - anularea redirectării unui dispozitiv de rețea.

2.96.1. Funcția 5F02h - Obținerea unei intrări în lista de redirectări

Subfuncția 02h returnează informații despre o anumită intrare în lista de redirectări a rețelei (creată cu funcția 5F03h), plasînd în două zone tampon rezervate de utilizator numele local și numele de rețea al dispozitivului.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 5Fh;
AL - încărcat cu 02h;
BX - încărcat cu indexul din lista de redirectări (prima intrare este zero);
DS:SI - încărcat cu un pointer la o zonă tampon de memorie de 16 octeți pentru numele local;

ES:DI - încărcat cu un pointer la o zonă tampon de memorie de 128 octeţi pentru numele de reţea.

Funcţia returnează următoarele:

. În AX - un cod de eroare, dacă după retur este poziţionat indicatorul CF:

1 - nu s-a încărcat programul de reţea;
18 - indexul din BX este mai mare decît numărul de intrări în listă.

. În BX - un fanion de stare al dispozitivului:

bit 0 = 0 - dispozitiv valid;
bit 0 = 1 - dispozitiv invalid.

. În BL - tipul dispozitivului:

3 - imprimantă;
4 - driver disc.

. În CX - parametrul poziţionat de utilizator cu funcţia 5F03h.

Observaţii:

1. Subfuncţia este disponibilă doar în versiunile DOS >= 3.1, şi necesită încărcarea programului de reţea IBM PC sau Microsoft.

2. Pentru a lista toate dispozitivele din reţea, se execută funcţia repetat, începînd cu indexul 0 în BX şi incrementînd BX cu 1 pînă cînd apare un cod de eroare 18 în AX.

3. Pentru informaţii suplimentare, a se vedea funcţiile 5F03h, 5F04h.

ATENŢIE!

Funcţia distruge regiştrii DX şi BP.

2.92.2. Funcţia 5F03h - Redirectarea unui dispozitiv de reţea

Subfuncţia 03h redirectează o imprimantă sau un driver de disc (dispozitivul sursă) la un director de reţea (dispozitivul destinaţie).

Pentru apelul funcţiei, se folosesc regiştrii:

- AH - încărcat cu 5Fh;
- AL - încărcat cu 03h;
- BL - încărcat cu tipul dispozitivului:
  - 3 - imprimantă;
  - 4 - driver de disc;
- CX - încărcat cu o valoare utilizator pe 16 biți, pe care o păstrează DOS (toate în afară de 0 sînt rezervate);
- DS:SI - încărcat cu un pointer la numele dispozitivului sursă (şir ASCIIZ);
- ES:DI - încărcat cu un pointer la numele dispozitivului destinaţie (şir ASCIIZ).

Dacă după retur este poziționat indicatorul CF, funcția returnează în AX un cod de eroare:

- 1 - nu s-a încărcat programul de rețea; valoarea din BX este eronată; șirul pentru sursă și/sau destinație este în format eronat; sursa este deja redirectată;
- 3 - director de rețea invalid;
- 5 - acces interzis;
- 8 - memorie insuficientă.

Observații:

1. Subfuncția este disponibilă doar în versiunile DOS >= 3.1, și necesită încărcarea programului de rețea IBM PC sau Microsoft.

2. Șirul destinație este de forma:

<nume_mașină><nume_cale><00h><parolă><00h>

unde:

nume_maşină - este numele server-ului care conţine directorul reţea;
nume_cale - este numele directorului de reţea la care este redirectată sursa;
parolă - este parola de acces la directorul de reţea. Dacă nu este specificată, cei doi octeţi nuli vor fi plasaţi imediat după nume_cale.

3. Dacă BL = 03h, numele acceptabile sînt PRN, LPT1, LPT2, LPT3 (de exemplu, "LPT1",0). În acest caz, redirectarea se face la nivelul INT 17 din ROM-BIOS.

4. Dacă BL = 04h, şirul sursă poate fi o literă de dispozitiv (de exemplu, "F:",0), sau un şir ASCIIZ nul. În acest din urmă caz, DOS încearcă să asigure accesul la numele reţea avînd parola specificată.

5. Pentru informaţii suplimentare, a se vedea funcţiile 5F02h, 5F04h, 59h.

2.96.3. Funcţia 5F04h - Anularea redirectării unui dispozitiv de reţea

Subfuncţia 04h anulează o redirectare realizată cu funcţia 5F03h, readucînd dispozitivul specificat la semnificaţia fizică iniţială.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 5Fh;
AL - încărcat cu 04h;
DS:SI - încărcat cu un pointer la numele dispozitivului sursă (şir ASCIIZ).

Dacă după retur este poziţionat indicatorul CF, funcţia returnează în AX un cod de eroare:

1 - nu s-a încărcat programul de reţea; sursă inexistentă;
15 - pauză în redirectare.

Observații:

1. Subfuncția este disponibilă doar în versiunile DOS >= 3.1, și necesită încărcarea programului de rețea IBM PC sau Microsoft.

2. Șirul ASCIIZ pointat de DS:SI poate conține una din următoarele 3 valori:
   - o literă de dispozitiv (de exemplu, "F:",0);
   - numele unei imprimante (de exemplu, "LPT1",0);
   - un șir începînd cu 2 caractere "\".

   În ultimul caz, se termină legătura între mașina locală și directorul de rețea.

3. Pentru informații suplimentare, a se vedea funcțiile 5F02h, 5F03h, 59h.

2.97. Funcția 60h - Rezervată

2.98. Funcția 61h - Rezervată

2.99. Funcția 62h - Obținerea adresei PSP-ului

Funcția 62h returnează adresa de segment a PSP-ului procesului curent.

Pentru apelul funcției, se folosește registrul:

AH - încărcat cu 62h.

Funcția returnează în BX adresa de segment a PSP-ului.

Observații:

1. Funcția este disponibilă doar în versiunile DOS >= 3.0.

2. Pentru informații suplimentare, a se vedea funcția 26h și PSP.

```
        mov     byte ptr fcb[0ah], 'x'
        mov     byte ptr fcb[0bh], 't'
;
        mov     ax, seg r               ;deschidere fişier
        mov     ds, ax
        mov     dx, offset fcb
        mov     ah, 0Fh
        int     21h
;
        cmp     al, 00h                 ;test eroare
        je      NoEr
;
        mov     ax, seg r               ;mesaj fişier negasit
        mov     ds, ax
        mov     dx, offset fnf
        mov     ah, 09h
        int     21h

        jmp     Finish
;
NoEr:
        mov     ax, seg r               ;mesaj gasire fisier
        mov     ds, ax
        mov     dx, offset ff
        mov     ah, 09h
        int     21h
;
;       afisare FCB
;
        mov     ax, seg r               ;mesaj afisare FCB
        mov     ds, ax
        mov     dx, offset dfcb
        mov     ah, 09h
        int     21h
;
        mov     ax, seg r               ;spatiere
        mov     ds, ax
        mov     dx, offset blanc
        mov     ah, 09h
        int     21h
```

```
        mov     ax, seg r               ;afisare unitate
        mov     ds, ax
        mov     ah, 02h
;
        mov     dl, byte ptr fcb[0]     ;conversie valoare ->
        add     dl, 30h                 ;caracter
        int     21h
;
        mov     dl, ':'
        int     21h
;
;       afisare nume prin citire in bucla
;
        mov     si, 1
Dname:
        mov     dl, byte ptr fcb[si]
        int     21h
        inc     si
        cmp     si, 9
        jne     Dname
;
        mov     dl, '.'
        int     21h
;
;       afisare extensie prin citire in bucla
;
Dext:
        mov     dl, byte ptr fcb[si]
        int     21h
        inc     si
        cmp     si, 12
        jne     Dext
;
        mov     dx, offset cr           ;<CR>
        mov     ah, 09h
        int     21h
;
        mov     ax, seg r               ;spatiere
        mov     ds, ax
        mov     dx, offset blanc
        mov     ah, 09h
```

```
        int     21h
;
;       afisare bloc curent prin citire si conversie
;
        mov     ax, seg r
        mov     ds, ax
        mov     si, 12
        mov     ax, word ptr fcb[si]
        mov     cl, 100
        div     cl
        mov     cl, ah
        add     al, '0'
        mov     dl, al
        mov     ah, 02h
        int     21h
        xor     ax, ax
        mov     al, cl
        mov     cl, 10
        div     cl
        mov     ch, ah
        add     al, '0'
        mov     dl, al
        mov     ah, 02h
        int     21h
        add     ch, '0'
        mov     dl, ch
        int     21h
;
        mov     dl, ' '                         ;blanc
        int     21h
;
;       afisare dimensiune articol prin citire si conversie
;
        mov     ax, seg r
        mov     ds, ax
        mov     si, 14
        mov     ax, word ptr fcb[si]
        mov     cl, 100
        div     cl
        mov     cl, ah
        add     al, '0'
```

```
        mov     di, al
        mov     ah, 02h
        int     21h
        xor     ax, ax
        mov     al, cl
        mov     cl, 10
        div     cl
        mov     ch, ah
        add     al, '0'
        mov     dl, al
        mov     ah, 02h
        int     21h
        add     ch, '0'
        mov     dl, ch
        int     21h
;
        mov     dx, offset cr           ;<CR>
        mov     ah, 09h
        int     21h
;
        mov     ax, seg r               ;inchidere fisier
        mov     ds, ax
        mov     dx, offset fcb
        mov     ah, 10h
        int     21h
;
        mov     ax, seg r               ;intrebare stergere
        mov     ds, ax
        mov     dx, offset qd
        mov     ah, 09h
        int     21h
;
        mov     ah, 01h                 ;citire raspuns
        int     21h
;
        cmp     al, 'd'                 ;test stergere
        jne     Finish
;
        mov     ax, seg r               ;stergere fisier
        mov     ds, ax
        mov     dx, offset fcb
```

```
        mov     ah, 13h
        int     21h
;
Finish:
        mov     ah, 4ch                     ;terminare program
        int     21h
;
r       ends
end     Start
```

Funcţia 65h - Obţinerea informaţiilor extinse dependente de ţară

Funcţia returnează, într-o zonă tampon prestabilită, informaţii dependente de ţară, în mod mai detaliat decît funcţia 38h.

AH - încărcat cu 65h;
AL - încărcat cu numărul operaţiei de executat:
  1- pentru obţinerea informaţiilor generale de internaţionalizare;
  2- pentru obţinerea pointerului la tabela de litere mari;
  4- pentru obţinerea pointerului la tabela de litere mari pentru nume de fişiere;
  6- pentru obţinerea pointerului la tabela de confruntare;
  7- pentru obţinerea pointerului la vectorul pentru setul de caractere pe doi octeţi (DBCS).
BX - încărcat cu numărul paginii de cod;
CX - încărcat cu lungimea zonei tampon pentru recepţionarea informaţiilor (minimum 5 octeţi);
DX - încărcat cu identificatorul de ţară (implicit -1);
DI - încărcat cu deplasamentul zonei tampon pentru recepţionarea informaţiilor;
ES - încărcat cu segmentul zonei tampon pentru recepţionarea informaţiilor;

Organizarea informaţiilor returnate în zona tampon:

Octet	Bit	Descriere
Date returnate de funcţia 01h		
00h		Identificatore de informaţie
01h - 02h		Lungimea tamponului care urmează
03h - 04h		Identificator de ţară
05h - 06h		Numărul paginii de cod
07h - 08h		Formatul datei 0 - SUA (L-Z-A) 1 - Europa (Z-L-A) 2 - Japonia (A-L-Z)

09h - 0Dh		Simbol monetar (şir ASCIIZ)
0Eh - 0Fh		Separator de mii (şir ASCIIZ)
10h - 11h		Separator zecimal (şir ASCIIZ)
12h - 13h		Separator de dată (şir ASCIIZ)
14h - 15h		Separator de oră (şir ASCIIZ)
16h	0	0 - Simbolul monetar precede valoarea
		1 - Simbolul monetar urmează după valoare
	1	0 - Nu există spaţiu între valoare şi simbolul monetar
		1 - Un spaţiu între valoare şi simbolul monetar
	2	0 - Simbolul monetar şi zecimalele sînt separate
		1 - Simbolul monetar înlocuieşte separatorul zecimal
17h		Numărul de zecimale în valoarea monetară
18h		Formatul orei (bit 1)
		0 - Ceas pe 12 ore
		1 - Ceas pe 24 de ore
19h - 1Ch		Pointer la o procedură de tip FAR
1Dh - 1Eh		Separator în lista de date (şir ASCIIZ)
1Fh - 28h		Rezervaţi

Date returnate de funcţiile 02h, 04h şi 06h

00h	Identificator de informaţie
01h - 05h	Pointer la tabel pe dublu cuvînt

Tabelele pentru litere mari şi cea de litere mari pentru nume de fişiere au o lungime de maximum 130 de octeţi, primii doi specificînd lungimea tabelei. Restul octeţilor conţin caracterele echivalente pentru codurile ASCII extinse 128 - 255, permiţînd înlocuirea caracterelor obişnuite cu cele cu accent utilizate în nume de fişiere şi documente. Utilizarea lor cu nume de fişiere permite accesarea fişierelor fără a utiliza o tastatură specială. De asemenea, ele pot fi utilizate la transformarea ieşirii pentru o imprimantă, întrucît unele imprimante nu permit setul ASCII extins IBM.

Tabela de confruntare conţine maximum 258 de octeţi, primii doi specificînd lungimea sa. Restul tabelei conţine caracterele care trebuie substituite caracterelor ASCII 0 - 255 pentru operaţiile

de sortare. Acest fapt permite sortări care nu deosebesc literele mari de cele mici şi aşează caracterele cu accent pe poziţiile corecte.

Dacă după retur este poziţionat indicatorul CF, funcţia returnează în AX un cod de eroare.

Pentru informaţii suplimentare, a se vedea funcţiile 38h, 66h.

Funcţia 66h - Obţinerea/Stabilirea paginii de cod

Această funcţie returnează sau selectează pagina de cod curentă. Citirea ei se face din fişierul COUNTRY.SYS.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 66h;
AL - încărcat cu:
  1- pentru returnarea paginii de cod;
  2- pentru returnarea paginii de cod.
BX - încărcat cu numărul paginii de cod (pentru AL = 2).

Dacă după retur este poziţionat indicatorul CF, funcţia returnează în AX un cod de eroare. În caz contrar, funcţia returnează următoarele (pentru AL = 1):

.în BX - numărul paginii de cod active;
.în DX - numărul paginii de cod implicite.

Pentru informaţii suplimentare, a se vedea funcţiile 38h, 65h.

Funcţia 67h - Stabilirea numărului de identificatori logici

Funcţia stabileşte numărul de identificatori logici disponibili simultan pentru deschiderea fişierelor şi dispozitivelor.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 67h;
BX - încărcat cu numărul de identificatori logici.

Dacă după retur este poziţionat indicatorul CF, funcţia returnează în AX un cod de eroare.

Această funcţie controlează dimensiunea tabelei utilizate de DOS pentru a memora identificatorii de fişiere, aceasta putînd memora implicit 20 de identificatori. Execuţia poate eşua dacă un program necesită deschiderea unui număr mai mare de fişiere iar sistemul nu posedă memorie suficientă pentru tabela extinsă.

Pentru informaţii suplimentare, a se vedea funcţiile 3Ch, 3Dh, 3Fh, 40h, 5Bh, 68h.

Funcţia 68h - Memorarea într-un fişier

Funcţia trimite orice date aflate curent în zone tampon DOS spre identificatorul logic specificat. Dacă acesta se referă la un fişier, este actualizată data şi ora ultimei modificări. În medii multitasking sau pentru lucrul în reţea, funcţia este preferabilă operaţiei de deschidere/închidere a unui fişier.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 68h;
BX - încărcat cu identificatorul logic.

Dacă după retur este poziţionat indicatorul CF, funcţia returnează în AX un cod de eroare.

Pentru informaţii suplimentare, a se vedea funcţiile 3Ch, 40h, 67h.

Anexa B. Modificări aduse de versiunea DOS 4.00

În versiunea DOS 4.00, unele funcţii ale întreruperii 21h au fost modificate/completate, fiind adăugate şi unele funcţii noi:

B1. Funcţii revizuite

Funcţia 33h - Poziţionarea/Obţinerea stării indicatorului CTRL/C, obţinerea unităţii de încărcare

Versiunea DOS 4.00 a introdus o acţiune nouă pentru această funcţie, în completare la cele deja existente: obţinerea unităţii de încărcare.

În acest ultim caz, pentru apelul funcţiei se folosesc regiştrii:

AH - încărcat cu 33h;
AL - încărcat cu 5h.

Funcţia returnează în registrul DL numărul unităţii de pe care a fost încărcat sistemul de operare (1 = A, 2 = B, etc.).

Funcţia 59h - Obţinerea informaţiilor suplimentare despre o eroare

În cadrul acestei funcţii, versiunea DOS 4.00 a introdus următoarele coduri de eroare suplimentare, majoritatea referindu-se la lucrul în reţea:

24h	Zonă tampon de partajare depăşită.
25h - 31h	Rezervate.
32h	Cerere reţea nepermisă.
33h	Maşină la distanţă nepregătită.
34h	Nume duplicat în reţea.
35h	Nume negăsit în reţea.
36h	Reţea ocupată.
37h	Dispozitivul de reţea nu mai există.
38h	Depăşirea limitei de comandă NetBIOS.
39h	Eroare în hardware-ul de adaptare al reţelei.
3Ah	Răspuns incorect de la reţea.

3Bh	Eroare de rețea neașteptată.
3Ch	Adaptor la distanță incompatibil.
3Dh	Coadă de listare plină.
3Eh	Spațiu insuficient pentru fișierul de listat.
3Fh	Listarea fișierului anulată.
40h	Nume rețea șters.
41h	Acces în rețea interzis.
42h	Tip dispozitiv rețea incorect.
43h	Nume rețea negăsit.
44h	Depășire limitei numelui de rețea.
45h	Depășirea limitei sesiunii NetBIOS.
46h	Pauză temporară în partajarea fișierelor.
47h	Cerere rețea neacceptată.
48h	Pauză în redirectarea imprimantei sau discului.
49h - 4Fh	Rezervate.
51h	Rezervată.
54h	Prea multe redirectări.
55h	Redirectare duplicată.
56h	Parolă invalidă.
57h	Parametru invalid.
58h	Eroare în dispozitivul de rețea.
59h	Funcție nepermisă în rețea.
5Ah	Componenta sistem cerută nu este instalată.

B2. Funcții nou adăugate

Funcția 6Ch - Deschiderea unui fișier - versiune extinsă

Funcția deschide, creează sau înlocuiește un fișier. Dacă nu se specifică altfel, se lucrează pe unitatea și în directorul curent.

Pentru apelul funcției, se folosesc regiștrii:

AH - încărcat cu 6Ch;
AL - încărcat cu 0.
BX - încărcat cu valoarea care specifică modul de acces;
CX - încărcat cu valoarea care specifică atributul;
DX - încărcat cu valoarea care specifică indicatorul de deschidere;

SI - încărcat cu deplasamentul unui şir ASCIIZ care specifică numele de fişier;
DS - încărcat cu segmentul unui şir ASCIIZ care specifică numele de fişier;

Modul de acces extins

Bit	Valoare	Descriere
0 - 2	000	Citire
	001	Scriere
	010	Scriere/Citire
3		Rezervat (0)
4 - 6	000	Mod compatibilitate
	001	Interzis orice
	010	Interzisă scrierea
	011	Interzisă citirea
	100	Interzis nimic
7	0	Procesul fiu nu moşteneşte identificatorul logic
	1	Procesul fiu moşteneşte identificatorul logic
8 - 12		Rezervaţi (0)
13	0	Execută întreruperea 24h
	1	Returnează procesului codul de eroare
14	0	Scriere prin zonă tampon
	1	Scriere fizică în momentul cererii
15		Rezervat (0)

Atributele

Bit	Valoare	Semnificaţie
0	1	Read-only
1	1	Ascuns
2	1	Sistem
3	1	Etichetă de volum
4		Rezervat (0)
5	1	Arhivă
6 - 15		Rezervaţi (0)

Indicatorul de deschidere

Bit	Valoare	Descriere
0 - 3		Acţiune dacă fişierul există
	0000	Eşec
	0001	Deschiderea fişierului
	0010	Înlocuirea fişierului
4 - 7		Acţiune dacă fişierul nu există
	0000	Eşec
	0001	Crearea fişierului
8 - 15		Rezervaţi (0)

Dacă după retur este poziţionat indicatorul CF, a apărut o eroare al cărei cod este returnat în AX. În caz contrar, funcţia returnează următoarele:

. în AX - identificatorul logic al fişierului;
. în CX - acţiunea efectuată:
  1 - deschiderea unui fişier existent;
  2 - crearea unui fişier nou;
  3 - înlocuirea unui fişier existent.

Execuţia funcţiei poate eşua din mai diverse motive: cale inexistentă, director plin, existenţa unui fişier cu acelaşi nume şi cu atributul 'read-only', programul nu are suficiente drepturi de acces, etc.

Anexa C. Funcţii nedocumentate

În documentaţiile oficiale DOS există numere de funcţii specificate ca fiind "Rezervate". O parte din acestea există însă, fiind chiar de o reală utilitate. De asemenea, există şi alte întreruperi, indicatori sau cîmpuri de structuri nedocumentate. Astfel, în PSP există următoarele cîmpuri nedocumentate:

Deplasament	Dimensiune	Conţinut
16h	Cuvînt	Adresa PSP-ului programului părinte
18h	20 octeţi	Tabela Fişierelor de Sarcini (JFT - Job File Table) implicită
32h	Cuvînt	Numărul maxim de fişiere deschise (dimensiunea JFT)
34h	Pointer FAR	Adresa JFT-ului curent al programului

Faţă de aceste aspecte, Microsoft adoptă o politică standard: "Ele nu sînt documentate întrucît nu putem oferi nici o garanţie despre existenţa lor în dezvoltările ulterioare ale DOS-ului. Dacă aflaţi despre ele şi începeţi să le utilizaţi, există şanse mari ca aplicaţiile să nu poată fi rulate sub viitoarele versiuni DOS."

Există însă şi alte puncte de vedere. Astfel, unele dintre aceste caracteristici sînt esenţiale pentru lucrul cu programe rezidente sau pentru scrierea unui software de reţea, fiind utilizate de o mulţime de programatori şi firme de software puternice.

O parte din aceste funcţii vor fi prezentate în continuare.

Funcţia 32h - Obţinerea adresei Blocului de Parametri a Discului

Funcţia returnează adresa Blocului de Parametri a Discului (DPB) pentru o unitate specificată.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 32h;
DX - încărcat cu numărul unităţii (0 = implicit, 1 = A, etc.).

Funcţia returnează în DS:BX adresa DPB.

DPB are următoare structură:

DOS 2x şi DOS 3.x

Deplasament	Lungime	Conţinut
00h	Octet	Numărul unităţii
01h	Octet	Unitate logică în cadrul driver-ului
02h	Cuvînt	Octeţi pe sector
04h	Octet	(Sectoare pe unitate de alocare) - 1
05h	Octet	Deplasare pentru calculul numărului de sectoare pe unitate de alocare
06h	Cuvînt	Numărul de sectoare rezervate pentru zona de încărcare
08h	Octet	Numărul de copii ale FAT-ului
09h	Cuvînt	Numărul maxim de intrări în directorul rădăcină
0Bh	Cuvînt	Primul sector de date
0Dh	Cuvînt	Numărul ultimei zone de alocare
0Fh	Octet	Sectoare pe FAT
10h	Cuvînt	Primul sector din director
12h	Dublu cuvînt	Adresa driverului de dispozitiv
16h	Octet	Octet descriptor pentru tipul unităţii
17h	Octet	Indicator de acces
18h	Dublu cuvînt	Pointer la următorul Bloc de Parametri
1Ch	Dublu cuvînt	Rezervat (în general FFFF:0000)

DOS 4x

Deplasament	Lungime	Conţinut
00h	Octet	Numărul unităţii
01h	Octet	Unitate logică în cadrul driver-ului
02h	Cuvînt	Octeţi pe sector
04h	Octet	(Sectoare pe unitate de alocare) - 1
05h	Octet	Deplasare pentru calculul numărului de sectoare pe unitate de alocare

06h	Cuvînt	Numărul de sectoare rezervate pentru zona de încărcare
08h	Octet	Numărul de copii ale FAT-ului
09h	Cuvînt	Numărul maxim de intrări în directorul rădăcină
0Bh	Cuvînt	Primul sector de date
0Dh	Cuvînt	Numărul ultimei zone de alocare
0Fh	Cuvînt	Sectoare pe FAT
11h	Cuvînt	Primul sector din director
13h	Dublu cuvînt	Adresa driverului de dispozitiv
17h	Octet	Octet descriptor pentru tipul unităţii
18h	Octet	Indicator de acces
19h	Dublu cuvînt	Pointer la următorul Bloc de Parametri
1Dh	Dublu cuvînt	Rezervat (în general FFFF:0000)

ATENŢIE!

Această funcţie este disponibilă în versiunile DOS 2.0 pînă la 4.01. Fiind nedocumentată, ar putea să nu fie permisă în versiunile DOS ulterioare.

Funcţia 34h - Obţinerea adresei indicatorului InDOS

Pentru programele rezidente există doi indicatori esenţiali:

-indicatorul InDOS - dacă acesta are valoarea 0, înseamnă că în momentul respectiv nu se execută nici o funcţie a întreruperii 21h;
-indicatorul pentru handler-ul de eroare critică - acesta se află la adresa precedentă faţă de indicatorul InDOS.

Funcţia incrementează indicatorul InDOS la fiecare apariţie a unei cereri pentru o funcţie a întreruperii 21h, şi îl decrementează la revenire. Întrucît DOS-ul nu este reentrant iar un program rezident poate apela o funcţie DOS, trebuie ştiut dacă programul executat curent utilizează o întrerupere DOS.

Pentru apelul funcţiei, se foloseşte registrul:

AH - încărcat cu 34h.

Funcţia returnează în ES:DI adresa indicatorului InDOS.

ATENŢIE!

Această funcţie este disponibilă în versiunile DOS 3.1 pînă la 4.01. Fiind nedocumentată, ar putea să nu fie permisă în versiunile DOS ulterioare.

Funcţia 50h - Stabilirea adresei PSP-ului

Funcţia se utilizează în programe rezidente (TSR), comunicînd DOS-ului să utilizeze adresa PSP-ului (Programm Segment Prefix) programului rezident în loc de adresa PSP-ului programul executat curent.

Pentru apelul funcţiei, se folosesc regiştrii:

AH - încărcat cu 50h;
BC - încărcat cu segmentul noului PSP.

ATENŢIE!

Această funcţie este disponibilă în versiunile DOS 3.1 pînă la 4.01. Fiind nedocumentată, ar putea să nu fie permisă în versiunile DOS ulterioare.

Funcţia 51h - Obţinerea adresei PSP-ului

Funcţia returnează adresa PSP-ului programul executat curent.

Pentru apelul funcţiei, se foloseşte registrul:

AH - încărcat cu 51h.

Funcţia returnează în registrul BX segmentul PSP-ului.

Funcţia se poate utiliza în cadrul programelor rezidente. Adresa PSP-ului programului nerezident trebuie salvată, iar apoi DOS-ului i se comunică utilizarea în locul acesteia a adresei PSP-ului programului rezident.

ATENŢIE!

Această funcţie este disponibilă în versiunile DOS 3.1 pînă la 4.01. Fiind nedocumentată, ar putea să nu fie permisă în versiunile DOS ulterioare. În locul ei se poate utiliza funcţia 62h.

Funcţia 5Dh - Stabilirea informaţiilor suplimentare despre o eroare

Funcţia stabileşte informaţiile returnate de următorul apel al funcţiei 59h. Regiştrii DS:DX conţin adresa unei structuri de date de 11 cuvinte. Primele 8 sînt returnate de DOS în regiştrii AX, BX, CX, DX, SI, DI, DS şi ES prin la apelul funcţiei 59h. Ultimele 3 sînt poziţionate pe 0.

Pentru apelul funcţiei, se folosesc regiştrii:

AX - încărcat cu 5D0Ah;
DX - încărcat cu deplasamentul structurii de 11 cuvinte.
DS - încărcat cu segmentul structurii de 11 cuvinte.

ATENŢIE!

Această funcţie este disponibilă în versiunile DOS 3.1 pînă la 4.01. Fiind nedocumentată, ar putea să nu fie permisă în versiunile DOS ulterioare. Ca şi în cazul funcţiei 59h, nu se recomandă scrierea unui cod de program care să recunoască un anumit mesaj de eroare, întrucît acestea sînt actualizate cu fiecare versiune DOS.



micro **informatica srl**

EDITURA MICROINFORMATICA

Str. Observatorului nr. 1, bl. OS1,
Cluj-Napoca, 3400
Oficiul P.T.T.R. Cluj-Napoca 1, C.P. 186
tel. 95/118263

Cartea de față este destinată programatorilor care doresc să utilizeze resursele interne ale sistemului de operare MS-DOS, versiunile 3.20 - 4.00. Lucrarea oferă o descriere detaliată a funcțiilor disponibile prin întreruperea 21h, așa numitele **Funcții sistem**.

Lei 150,–